1984

Representation of potential flow about axisymmetric bodies with discrete singularities.

Janikowsky, Linda Crockett

Monterey, California. Naval Postgraduate School

NAVAL POSTGRADUATE SCHOOL
Monterey, California

THESIS

REPRESENTATION OF POTENTIAL FLOW ABOUT
AXISYMMETRIC BODIES WITH DISCRETE SINGULARITIES

by

Linda Crockett Janikowsky

March 1984

Thesis Advisor: T. Sarpkaya

Approved for public release; distribution unlimited.

REPORT DOCUMENTATION PAGE

READ INSTRUCTIONS BEFORE COMPLETING FORM

1. REPORT NUMBER	2. GOVT ACCESSION NO. / 3. RECIPIENT'S CATALOG NUMBER
4. TITLE (and Subtitle): Representation of Potential Flow About Axisymmetric Bodies with Discrete Singularities	5. TYPE OF REPORT & PERIOD COVERED: Master's Thesis; March 1984
	6. PERFORMING ORG. REPORT NUMBER
7. AUTHOR(s): Linda Crockett Janikowsky	8. CONTRACT OR GRANT NUMBER(s)
9. PERFORMING ORGANIZATION NAME AND ADDRESS: Naval Postgraduate School, Monterey, California 93943	10. PROGRAM ELEMENT, PROJECT, TASK AREA & WORK UNIT NUMBERS
11. CONTROLLING OFFICE NAME AND ADDRESS: Naval Postgraduate School, Monterey, California 93943	12. REPORT DATE: March 1984
	13. NUMBER OF PAGES: 160
14. MONITORING AGENCY NAME & ADDRESS(if different from Controlling Office)	15. SECURITY CLASS. (of this report): Unclassified
	15a. DECLASSIFICATION/ DOWNGRADING SCHEDULE

16. DISTRIBUTION STATEMENT (of this Report)

Approved for public release; distribution unlimited.

17. DISTRIBUTION STATEMENT (of the abstract entered in Block 20, if different from Report)

18. SUPPLEMENTARY NOTES

19. KEY WORDS (Continue on reverse side if necessary and identify by block number)

Potential Flow	Hydrodynamics
Axisymmetric Bodies	Numerical Body Simulation
Singularity Methods	Source-Sink Methods

20. ABSTRACT (Continue on reverse side if necessary and identify by block number)

A rational methodology has been developed whereby three-dimensional sources and sinks may be placed along the major axis of a class of ovary ellipsoids so as to minimize normal velocity and to calculate as exactly as possible the tangential velocity, pressure distribution, and the body shape. For this purpose the strength and position of the singularities and the position and number of the control points were optimized through the use of the method of least squares and the Automated Design

Synthesis optimization technique. The results have shown that the previous methods are far from satisfactory and the use of two types of optimization in the determination of the strength and position of the singularities yields and the desired body shape and flow characteristics with excellent accuracy. A comprehensive computer code has been developed to enable one to calculate most of the practically significant body shapes.

Approved for public release; distribution unlimited.

Representation of Potential Flow About Axisymmetric Bodies with Discrete Singularities

by

Linda Crockett Janikowsky
Lieutenant, United States Navy
B.S., George Peabody College for Teachers, 1977

Submitted in partial fulfillment of the
requirements for the degree of

MASTER OF SCIENCE IN MECHANICAL ENGINEERING

from the

NAVAL POSTGRADUATE SCHOOL
March 1984

ABSTRACT

A rational methodology has been developed whereby three-dimensional sources and sinks may be placed along the major axis of a class of ovary ellipsoids so as to minimize normal velocity and to calculate as exactly as possible the tangential velocity, pressure distribution, and the body shape. For this purpose the strength and position of the singularities and the position and number of the control points were optimized through the use of the method of least squares and the Automated Design Synthesis optimization technique. The results have shown that the previous methods are far from satisfactory and the use of two types of optimization in the determination of the strength and position of the singularities yields the desired body shape and flow characteristics with excellent accuracy. A comprehensive computer code has been developed to enable one to calculate most of the practically significant body shapes.

TABLE OF CONTENTS

LIST OF FIGURES

TABLE OF ABBREVIATIONS AND SYMBOLS

a	One half of the length of the major axis of an ellipsoid
a/b	Slenderness ratio
b	One half of the length of the minor axis of an ellipsoid
C_p	Pressure-coefficient
c	A constant; $c = \frac{a}{\cosh \xi}$.
E^2	Sum of the squares of the errors
J	Jacobian of the transformation
K	A constant [see Eq. 13]
m_i	$m_i = \frac{Q_i}{4\pi}$
N_c	Number of control points
N_{cc}	Number of control points after a given iteration
N_{eck}	Number of extra control points apportioned to the k-th interval
N_s	Number of singularities
p	Local pressure
p_∞	Ambient pressure
Q_i	Strength of the i-th singularity
q_η	Normal velocity component in elliptic coordinates
q_ξ	Tangential velocity component in elliptic coordinates
R_j	Error at the j-th control point
U	Free stream velocity (assumed to be U=1)
u	Velocity component in the x-direction

V_n	Velocity component normal on the body surface
V_t	Velocity component tangential on the body surface
v	Velocity component in the ω-direction
x	Linear distance along the major axis of the body
z	Complex variable
ζ	Complex variable in elliptic coordinates
η	Imaginary part of ζ, $\zeta = \xi + i\,\eta$
θ	An angle (see Fig. 2)
ξ	Real part of ζ
ρ	Density of fluid
Σ	Total sum of the absolute values of the normal velocities along the length of the body
ψ	Stokes' stream function
ω	Radial distance from the major axis to the body surface

ACKNOWLEDGEMENT

The author wishes to express her deep appreciation to Distinguished Professor Turgut Sarpkaya for his guidance, instruction, and support which made the completion of this project possible. It has been an honor to study with a man of Professor Sarpkaya's professional stature. To work with this individual is an opportunity to grow not only professionally but also personally.

Many thanks to my colleagues and friends who were genuinely sympathetic in the moments of frustration that are part of any thesis project.

Most of all, deepest thanks to my husband, Ralph, whose unfailing love and support made the work and sacrifice bearable.

I. INTRODUCTION

The development of reliable methods for the calculation of three-dimensional viscous flows on shapes of hydrodynamic interest requires the analysis of (a) the inviscid flow, (b) three-dimensional laminar and turbulent boundary layers, (c) flow separation, (d) post-separation flow and wake, and (e) the interaction between the viscous and inviscid flow regions.

A critical survey of the present state of the art indicates that many aspects of the problem remain tentative or unresolved even though considerable progress has been made in recent years in improving the calculation methods for inviscid flow and boundary layers about three-dimensional bodies. Further progress in each of these areas is hampered largely by the lack of pertinent data from carefully conceived and executed experiments. To be sure, such experiments are difficult and very expensive. The availability of high-speed computers, on the other hand, has led to the development of novel calculation procedures which claim a level of generality that has far surpassed the reliability of the underlying assumptions and techniques. In the present study, attention is focused on the inviscid flow about axisymmetric bodies and its calculation through the use of exact and approximate methods of optimization to the desired degree of accuracy.

The potential flow about an arbitrary body of revolution was first treated by von Karman [1]. He determined the potential flow around bodies of revolution at zero angle of attack by superposing a uniform

stream on a system of line sources distributed along the axis of the body. An equal number of coordinates (control points) were chosen on the body and the strengths of the sources were determined so that the zero streamline passed through the chosen coordinates.

Since the pioneering work of von Karman, very little original work has appeared in the literature. While most fluid dynamicists are familiar with the method of representation of a body with distributed sources (sinks) and surface panels, they generally do not recognize the limitations of the methods and provide no guidance as to how one can systematically approach the ideal solution. Often, broad statements and vague suggestions are made regarding the slenderness of the body, the number of the singularities (i.e., sources and sinks), and the magnitude of the differences between the calculated and exact solutions.

Another approach to solving the inviscid flow problem for bodies of revolution is the surface singularity or surface panel method. This approach is equivalent to the solution of an integral equation (Fredholm integral of the second kind). However, the computational effort in the axial-singularity-distribution method of von Karman is a fraction of that of the surface singularity methods. Furthermore, the surface singularity methods are not necessarily more accurate than the methods of discrete or continuous singularities along the axis.

It is, therefore, necessary to define more precisely the number of singularities, their strengths and positions, the number of control points, the differences between the calculated and given body shape, the deviation of the normal velocity from zero on the body surface, the discontinuities or artificial oscillations in the tangential velocity and in the pressure

distribution on the body surface, and as to how one can assess the degree of accuracy of the calculation procedures and improve systematically the accuracy of the quantities calculated. This investigation deals with these questions and significantly improves the calculation of the incompressible potential flow about bodies of revolution at zero angle of attack.

II. PREVIOUS STUDIES

Von Karman [1] in 1927 used a method involving line sources clustered in the bow and line sinks clustered in the stern of an airship. The method of solution separated the bow computations from the stern computations since the influence of one on the other was considered small. In this method a good comparison of airship pressures with experiment was obtained. However, for shapes more complex than airships one would have to consider sources and sinks along the whole axis of the body and one would not isolate discrete parts of the body for separate computations. In doing so, however, one discovers that the method is unstable or may fail unexpectedly.

Hess [2] and Oberkampf and Watson [3] have critically examined the generalizations of von Karman's method. They have shown that the method produces a system of linear equations which is, in general, ill-conditioned, and requires very high computational accuracy in the construction of the coefficient matrix and in solving the equations. Oberkampf and Watson concluded that the method does not always produce reliable solutions for the flow around a specified body and that the conditions that the body should meet in order to be represented by a system of axial line sources are not clear. Karamcheti [4] states that the body should be slender and should not have any discontinuities in the slope of the meridian line. Numerical experiments by Oberkampf and Watson showed that such conditions are not sufficient. For example, the method gave a slight rippling in the velocity distribution as well as in the meridian

streamline of very slender Rankine ovals. Oberkampf and Watson attributed the rippling in velocity to the increased local effects of each source element. Another result of their study is that the method is sensitive not only to the shape of the body contour but also to the number of elements used to generate the body. Oberkampf and Watson attempted to calculate the potential flow about a sphere using an odd number of sources vice an even number. The results for an odd number of sources were very peculiar. While the zero streamline did pass through all of the specified coordinates of the sphere, the body generated had "holes" in its surface. That is, between the specified coordinates the zero streamline plunged into the negative strength line sources on the axis and then reappeared so that it passed through the next specified coordinate. This behavior produced highly unrealistic normal and tangential velocities. It is hypothesized that this erratic behavior was caused by the fact that with an odd number of line sources one of the sources must overlap the center of the sphere to preserve symmetry and, consequently, prohibits any antisymmetric solution. Therefore, odd numbers of sources must be avoided for bodies that are symmetric about a plane normal to their major axis at the midpoint of their length.

It is evident from the foregoing that von Karman's method does not always produce reliable solutions for the potential flow around a specified body. The alternative is to use surface singularities or panels [2, 4]. The panel methods could calculate all the required viscous and inviscid flow properties without further interaction on the part of the user once the geometry of the body and the flow conditions are specified. However, such methods require far more computation time than any other inviscid flow method. Accordingly, many investigators have concentrated

on representation of the inviscid flow through the use of discrete singularities (three-dimensional point sources and sinks) placed on the axis of the body. Once the surface velocity and pressure are calculated, the potential flow and the boundary layer are "patched together" to represent the real flow. It is important to note that in all of these calculations, regardless of the method of representation of the inviscid flow, it is assumed that the flow is essentially unseparated and the effects of viscosity are appreciable only in a very thin layer adjacent to the body surface and in a thin wake downstream of the body.

The ultimate goal of the use of discrete singularities along the axis of the body is to determine a flow which is, in some sense, a good approximation to the exact flow about the given body. If this approach is adopted, it is essential to know, however, the degree of approximation. This can be checked in a number of ways after one solves for the strength and/or the position of the singularities. One laborious way of checking the accuracy would be to increase the number of singularities, solve again for the strengths, positions, and velocities, and compare the two solutions for convergence. This process has three essential drawbacks. First, it can lead to inefficient use of the computer facilities. The computer time required for the solution increases rapidly with the number of singularities. Thus, the cost for solution depends very strongly on the number of the singularities used. If the number of singularities is increased until little change in the local velocities on the body is observed, then certainly the final and by far the most expensive computation (i.e., the last one) is redundant. Second, as discussed in the foregoing, such a process may not converge when distributed singularities

are used. Third, and perhaps most important, there is nothing in this process which yields an insight into how one might improve the strength and position of the singularities. In other words, it is difficult to determine if a more efficient computation would result if one region of the axis had more singularities and control points per unit length.

An alternative method of checking the accuracy of the solution is to evaluate the normal velocities on the exact surface of the body (i.e., on the shape specified rather than on the one represented by the zero streamline) at points other than the control points. If the body is nonporous, as assumed here, then these normal velocities should be identically zero. It is easily shown that the magnitude of the normal velocities on the exact body is, in fact, a proper measure of the inaccuracies in the whole flow field. Thus, the remnant normal velocity distribution at the end of a particular step may be used to improve the calculations and hence the calculated flow field.

In view of the simplicity of the use of discrete sources and sinks along the axis of the body as compared to the distributed axial singularities or surface singularity distributions, and the tremendous saving in the amount of memory storage and numerical calculation, it was considered justifiable to undertake an extensive study of the body representation by discrete singularities. In this study, improvement is realized by allowing the strength and position of the singularities and the number and positions of the control points, respectively, to achieve their optimum values.

III. MATHEMATICAL DESCRIPTION

In this section, the basic equations for a discrete axial distribution of three dimensional sources and sinks combined with uniform flow are derived. In addition, the exact tangential velocity for an ovary ellipsoid is obtained for comparison with that obtained numerically.

Stokes' stream function for an ambient flow of unit velocity past an axisymmetric body with N sources (sinks) of strength Q_i along its axis is given by [5]

$$\psi = \frac{1}{2}\omega^2 - \sum_{i=1}^{N} \frac{Q_i}{4\pi} \frac{x - x_i}{r} \tag{1}$$

where $r^2 = \omega^2 + (x - x_i)^2$ at the point (x,ω) in the flow field. Evidently, $\psi = 0$ corresponds to the enclosed body (see Fig. 1).

The velocity components u and v are given by

$$u = \frac{1}{\omega} \frac{\delta\psi}{\delta\omega} = 1 + \sum_{i=1}^{N} m_i \frac{x - x_i}{r^3} \tag{2}$$

$$v = -\frac{1}{\omega} \frac{\delta\psi}{\delta x} = \sum_{i=1}^{N} m_i \frac{\omega}{r^3} \tag{3}$$

where $m_i = Q_i/4\pi$.

The body is assumed to be defined either by a function $\omega = f(x)$ or by a discrete set of points. In either case, there is sufficient information to calculate the normal and tangential components of the velocity along the body. In fact, from Fig. 2 one has

$$V_n = v \cos\theta - u \sin\theta \tag{4}$$

$$V_t = v \sin\theta + u \cos\theta \tag{5}$$

where θ is defined by

$$\tan\theta = df/dx \tag{6}$$

It is evident from the foregoing that the accurate specification of the stream function determines the accuracy of the tangential velocity and the existence of non-zero normal velocities along a non-porous wall indicates the error between the calculated and ideal solution.

The numerical experiments have been carried out with ellipsoids of various a/b ratios (see Fig. 1). Thus, it was necessary to derive the exact expression for the tangential velocity so that the accuracy of the numerical method may be properly assessed.

The transformation

$$z = x + i\omega = c \cosh\zeta \tag{7}$$

yields

$$x = c \cosh\xi \cos\eta \tag{8}$$

and

$$\omega = c \sinh\xi \sin\eta \tag{9}$$

Thus, $\xi = \xi_o$ denotes an ellipse, in the meridian plane, as

$$\frac{x^2}{(c \cosh\xi_o)^2} + \frac{\omega^2}{(c \sinh\xi_o)^2} = 1 \tag{10}$$

whose semi-axes are

$$a = c \cosh\xi_o \quad , \quad b = c \sinh\xi_o \tag{11}$$

Note that $b/a = \tanh\xi_o$ determines ξ_o.

It is relatively easy to show that the stream function is given by

$$\psi = \frac{1}{2} c^2 \sinh^2\xi \sin^2\eta - \frac{1}{2} \frac{b^2}{K^2} \left(\cosh\xi + \sinh^2\xi \cdot \text{Ln} \tanh \frac{\xi}{2}\right) \sin^2\eta \qquad (12)$$

in which

$$K = \frac{a}{c} + \frac{b^2}{c^2} \text{Ln} \frac{a + b - c}{a + b + c} \qquad (13)$$

The velocity components are given by [6]

$$q_\xi = - \frac{1}{J\omega} \frac{\delta\psi}{\delta\eta} \quad , \quad q_\eta = \frac{1}{J\omega} \frac{\delta\psi}{\delta\xi} \qquad (14)$$

where

$$\zeta = \xi + i\eta \qquad (15)$$

and

$$J^2 = f'(\zeta) \overline{f'(\zeta)} \qquad (16)$$

and

$$f(\zeta) = c \cosh\zeta \quad \text{and} \quad f'(\zeta) = c \sinh\zeta \qquad (17)$$

Equations (16) and (17) yield

$$J = \frac{c}{\sqrt{2}} (\cosh 2\xi - \cos 2\eta)^{\frac{1}{2}} \qquad (18)$$

Thus, the velocity components on the body (i.e., for $\xi = \xi_o$) are

$$q_\xi = - \frac{\sqrt{2} \cos\eta}{\sqrt{\cosh 2\xi_o - \cos 2\eta}} \left[\sinh\xi_o - \frac{b^2}{Kc^2} (\text{ctnh}\xi_o + \sinh\xi_o \cdot \text{Ln} \tanh \frac{\xi_o}{2})\right] \qquad (19)$$

and

$$q_\eta = \frac{\sqrt{2}\sin\eta}{\sqrt{\cosh 2\xi_o - \cos 2\eta}} \left[\cosh\xi_o - \frac{b^2}{Kc^2}\left(1 + \cosh\xi_o \cdot \mathrm{Ln}\tanh\frac{\xi_o}{2}\right)\right] \qquad (20)$$

It can be shown that Eq. (19) is identically zero since it represents the normal velocity on the body. The tangential velocity is calculated from Eq. (20) for representative values of a/b and is shown in Figs. 3a and 3b as a function of the normalized distance from the forward stagnation point.

IV. NUMERICAL ANALYSIS AND RESULTS

A. USE OF EQUAL NUMBER OF SINGULARITIES AND CONTROL POINTS

As noted in connection with the discussion of previous investigations, the use of the singularity methods requires the selection of the position of the singularities along the axis and of the control points on the body. A convenient but arbitrary selection of boundary points and singularity locations in this procedure is to position the singularities directly below the boundary points in one-to-one correspondence [6]. Evidently, no special criteria is provided for the spacing or number of the singularities. In any case, this procedure results in a set of linear equations which can be solved through the use of standard matrix reduction techniques.

The first example chosen to illustrate the technique and the problems associated with it is an ovary ellipsoid (also called a prolate spheroid, generated by rotation of an ellipse about its major axis). The ellipsoid had a slenderness ratio of a/b = 6.0. An even number of singularities (N_s) was chosen and they were equally spaced along the major axis of length 2a. An equal number of control points (N_c) was placed directly above the singularities on the ellipse as shown in Fig. 4.

Figures 5a through 5f show the results obtained with four singularities and four control points. Figure 5a shows the normal component of the velocity (V_n) calculated along the upper half of the ellipse. The condition of $\psi = 0$ is exactly satisfied at the control points. However, large non-zero normal velocities between the control points show clearly

the inadequacy of the body representation by four equally spaced singularities and control points. Figure 5b shows the tangential velocity (V_t) along the body. Theoretically, one would expect a smoothly increasing tangential velocity profile. Thus, the large oscillations in V_t are a further indication of the failure of only four singularities and control points, as presently positioned, to adequately represent the body. Figure 5c shows a comparison of the theoretical and calculated V_t as a function of the normalized distance from the forward stagnation point while Fig. 5d depicts the difference between theoretical and calculated tangential velocity along the first half of the body. Apparently, the calculated V_t oscillates about the exact velocity profile. In Fig. 5e the pressure-coefficient C_p, [$C_p = (p - p_\infty)/(0.5\ U^2)$], and in Fig. 5f the body shape resulting from the 4/4 case[1] are presented. It is evident from the foregoing that an ellipsoid of a/b = 6.0 cannot be successfully represented by a small number of equally spaced singularities and control points. Thus, it is necessary to explore first the effect of increasing the number of singularities in the representation of the body and flow characteristics prior to embarking on a more ambitious investigation of the effect of using nonuniform singularity distributions.

In anticipation of improvement of the flow characteristics with an increased number of singularities and control points the cases of 10/10 and 20/20 were investigated. Figures 6a through 6f and Figs. 7a through 7f show the results for the cases 10/10 and 20/20, respectively.

---

[1] For convenience, the designation N_s/N_c, e.g., 4/4, will be used hereafter to refer to the number of singularities and control points used in a particular example.

Evidently, the difference between the theoretical and calculated tangential velocity, as well as, all other errors in the calculated flow characteristics have been reduced by increasing the number of singularities and control points. Nevertheless, the representation of the flow characteristics is far from satisfactory and one cannot use the resulting stream function to predict the boundary layer characteristics over the body. It is, therefore, necessary to explore other methods which will minimize the number of the singularities as well as the error between the predicted and calculated flow characteristics.

B. THE METHOD OF LEAST SQUARES

The use of an equal number of singularities and control points resulted in a deterministic set of linear equations and in making ψ exactly equal to zero at each and every control point., However, one could use a larger number of control points than singularities. This will obviously result in an over-determined set of linear equations if one attempted to render $\psi = 0$ at all control points. Recognizing the impossibility of doing so, one can, instead, minimize ψ at all control points through the use of a suitable minimization technique. Thus, one can make the error in ψ nearly zero at a larger number of control points in lieu of making it exactly zero at a fewer number of control points.

Let us consider N_s singularities and N_c control points. The stream function at the control point j due to the contributions of all singularities is given by (see Eq. (1)),

$$\psi_j = \frac{1}{2}\ \omega_j{}^2 - \sum_{i=1}^{N_s} m_i \left(\frac{x_j - x_i}{r}\right) \qquad (21)$$

If ψ were not exactly zero at the point j, then the error in ψ_j would be,

$$R_j = \sum_{i=1}^{N_s} m_i \left(\frac{x_j - x_i}{r}\right) - \frac{1}{2}\ \omega_j{}^2 \tag{22}$$

Then the sum of the square of the errors at all control points becomes,

$$E^2 = \sum_{j=1}^{N_c} R_j^2 \tag{23}$$

in which E^2 is a function of the strengths of N_s number of singularities only. The total error may be minimized through the use of Gauss' method [7] by writing the partial derivative of E^2 with respect to each m_i equal to zero, i.e.,

$$\frac{\partial E^2}{\partial m_i} = 0 \tag{24}$$

Performing the said analysis, one has

$$A^T A\ \underset{\sim}{m} = A^T b \tag{25}$$

which represents a matrix of $N_s \times N_s$ linear equations where

$$[A] = \left[\frac{x_j - x_i}{r}\right] \tag{26}$$

$$[b] = \left[\frac{1}{2}\ \omega_j{}^2\right] \tag{27}$$

and

$$\underset{\sim}{m} = m_i \tag{28}$$

In the foregoing, i varies from 1 to N_s and j, from 1 to N_c.

The least squares method described above was applied to the ellipsoid with a/b = 6.0 for the cases 4/11, 10/21, and 20/41. As before the singularities were equally spaced along the major axis. The control points were likewise located on the body with equal spacing. The results are shown in Figs. 8a through 8f for the case of 4/11, in Figs. 9a through 9f for the case of 10/21, and in Figs. 10a through 10f for the case of 20/41. The comparison of these figures with those cited earlier (i.e., with those having identical letter designations such as 5a and 8a, etc.) shows that the increase in the number of control points does in fact improve the calculated tangential velocity and reduce normal velocity. One can also observe some improvement in the body representation as evidenced by a comparison of Figs. 5f and 8f or of Figs. 7f and 10f. It is also clear that an ellipsoid of slenderness ratio of a/b = 6.0 cannot be adequately represented with four singularities even with a larger number of control points. Furthermore, even with a significantly larger number of singularities and control points (as in the case of 20/41) the method of least squares does not yield a satisfactory representation of the flow characteristics. This leads one to the conclusion that optimizing the strengths of the singularities alone is not sufficient to adequately model the flow about the body.

The realization of this fact leads to the hypothesis that the locations of the singularities and the control points may be less than optimum. To test this hypothesis, one could judiciously position both the singularities and the control points in the areas of the body where the error is large (i.e., large normal velocities) and then compare the calculated flow characteristics with those obtained in the case of evenly spaced

singularities and control points. Previous calculations have shown that the error in all flow characteristics is largest where the radius of curvature of the body is relatively small (i.e., regions near the fore and aft stagnation points). With this in mind, the two singularities closest to the center of the body in the case of 20/41 were moved to the new positions of $x/b = \mp 5.71429$ while the remainder of the singularities were kept in their original evenly spaced positions (see Fig. 1). The four control points closest to the center of the body were moved to new positions directly above $x/b = \mp 5.90476$ and $x/b = \mp 5.80952$ while the remainder of the control points were kept in their original positions. Figures 11a through 11f show the results obtained in this manner. A comparison of these Figures with Figs. 10a through 10f shows that V_n has been significantly reduced and there is greater agreement between the theoretical and calculated tangential velocities as a result of the repositioning of the singularities and the control points. Thus, one must devise a rational method with which the singularities and the control points can move to their optimum positions as the singularities continue to adjust their strengths, as in the previous cases, through the use of the method of least squares.

C. DUAL OPTIMIZATION METHODOLOGY

In the following, a method is described whereby the position and strength of the singularities and the number and position of the control points are progressively optimized so as to minimize the error in the prediction of the flow characteristics about a given body. For this purpose the following steps have been developed:

1. Decide on the appropriate number of singularities, N_s, for the body of given slenderness ratio (two singularities per unit length are recommended on the basis of the experience gained with the least squares method).

2. Position the singularities with equal spacing along the major axis of the body.

3. Select twice as many control points, N_c, as singularities and place them on the body contour with equal horizontal spacing.

4. Determine the strength of the singularities through the use of the method of least squares, as described previously.

5. Calculate the sum of the absolute values of the normal velocities between singularities N_i and N_{i+1} for i from 1 to N_s, as well as, in the regions between the forward stagnation point and the first singularity and between the last singularity and the rear stagnation point. This yields (N_s+2) sums corresponding to the (N_s+2) intervals. In addition, calculate the total sum of the absolute values of the normal velocities along the full length of the body (hereafter referred to as Σ).

6. Initially, place one control point per interval on the body contour. This requires $(N_s + 2)$ control points. The remaining $[N_c - (N_s + 2)]$ control points are apportioned among the intervals using the relative magnitude of the sum of the absolute values of the normal velocities in each interval as a weighing factor. Specifically, the extra control points are assigned in accordance with

$$N_{eck} = \frac{\sum_{k}^{1} |V_{n_x}| \Delta(\frac{x}{a})}{\sum_{-1}^{1} |V_{n_x}| \Delta(\frac{x}{a})} [N_{cc} - (N_s + 2)] \qquad (29)$$

where N_{cc} represents the number of control points at the end of a given iteration and k varies from 1 to (N_s+2). As progressive iterations of this process reduce the normal velocities along the body, less than the original number of control points may be necessary to adequately represent the body and the flow characteristics. Equation (29) allows for this and will reduce the total number of control points as needed. However, the number of control points will never be less than (N_s+2).

7. Place in each interval the apportioned number of control points with equal horizontal spacing.

8. Determine the improved positions for the N_s singularities through the use of the Automated Design Synthesis optimization program [8].

9. Repeat steps 4 through 8 until no further reduction is realized in Σ .

The complete computer program based on the steps described above is presented in Appendix A.

Several representative calculations have been performed to demonstrate the effectiveness of the dual optimization methodology. The first case concerns the ellipsoid with the slenderness ratio of a/b = 6.0. Twenty singularities and 40 control points were chosen in accordance with the suggestions made in Steps # 1 and 3. At the end of the first iteration (i.e., Steps #1 through 9) the total number of the control points needed was reduced to 36. Figures 12a through 12f show the results in graphical form. Clearly, the normal velocities are fairly large and the difference between the theoretical and calculated tangential velocity is not yet satisfactory. The computer program carried out a total number of 28 iterations at the conclusion of which Σ was reduced to its minimum value.

After the final iteration, the total number of the control points was reduced to 22. The control points and singularities became unevenly spaced and moved towards the fore and aft regions of the body. Figures 13a and 13b show, respectively, the initial and final positions of the singularities and control points. Figures 14a through 14f show that the difference between the theoretical and calculated tangential velocity is reduced to about 0.10xU very near the stagnation points and to less than about 0.04xU along the remainder of the body. Furthermore, the calculated velocities over the central half of the body are almost identical with those predicted theoretically.

It was noted earlier that the dual optimization program should begin with two singularities per unit length. To validate the significance of this suggestion and to provide another test case for the computer code, the case of 10/20 was considered for the ellipsoid of slenderness ratio of a/b = 6.0. The dual optimization program performed as expected and at the end of 22 iterations achieved the minimum Σ . In the course of the optimization, the number of the control points was reduced to 12. The final results are shown in Figs. 15a through 15f. A comparison of these figures with Figs. 14a through 14f shows that the use of less than two singularities per unit length does not produce flow characteristics as good as those calculated in the case of 20/22. Thus, the use of about two singularities per unit length is considered optimum. Obviously, it is always possible to use a larger number of singularities. However, the additional computing expense is not commensurate with the minimal improvement achieved in the flow characteristics.

To evaluate the ability of the computer code to deal with ellipsoids of other slenderness ratios, a test case with a/b = 2.0 was run. Initially, 8 singularities and 16 control points were equally spaced along the body. Only three iterations were required to achieve the minimum value of Σ. The results, presented in Fig. 16a through 16f, show that the difference between the calculated and theoretical velocities has been reduced to about 0.04xU along the entire body. Note also that the number control points required was reduced to 10 during the dual optimization process.

To further establish the generality of the dual optimization methodology, axisymmetric bodies without fore and aft symmetry have been considered. For bodies of this type there is no simple boundary function in the form of $\omega = f(x)$. The body shape is in general provided by the naval architect in accordance with the needs of the user. In the present study such a body has been generated through the use of one source and four sinks. The strengths and the positions of the sources and sinks were such that the resulting body did not have fore and aft symmetry. The Stokes stream function is given by

$$\psi = \frac{1}{2}\omega_i^2 - \sum_{i=1}^{5} m_i \frac{(x-x_i)}{[(x-x_i)^2 + \omega_i^2]^{1/2}} \qquad (30)$$

where

$m_1 = 0.24$ $\quad x_1 = 0.48891$

$m_2 = -0.06$ $\quad x_2 = 5.98891$

$m_3 = -0.06$ $\quad x_3 = 7.48891$

$m_4 = -0.06$ $\quad x_4 = 8.98891$

$m_5 = -0.06$ $\quad x_5 = 10.48891$

The resulting body shape is shown in Fig. 17. The exact tangential velocity calculated through the use of Eq. (30) is shown in Fig. 18. It will be compared later with that obtained numerically.

The optimization process followed the steps outlined previously. Thus, in accordance with the recommendation that about two singularities per unit length be chosen, 20 singularities and 41 control points were selected and distributed appropriately along the body. In only two iterations the number of control points reduced to 32 and Σ acquired its minimum value.

Figures 19a through 19f show the results after the final iteration. Clearly, the calculated and the exact body shapes are nearly identical and the normal velocity has been reduced to almost zero along the full length of the body. Also, the calculated and theoretical tangential velocities compare exceedingly well. It should be noted that the undulations in the tangential velocity at the aft end of the body are due to the geometry of the body (as defined by Eq. (30)) and are not attributable to any theoretical or numerical instability. In fact, the occurrence of such undulations in the exact tangential velocity profile provided greater challenge for the dual optimization process. The computer program for the case under consideration is presented in Appendix B.

It is evident from the foregoing that the dual optimization methodology can be used with great confidence in the prediction of the flow characteristics about axisymmetric bodies with or without fore and aft symmetry.

V. CONCLUSIONS

The investigation described herein warranted the following conclusions:

1. The existing methods for the representation of axisymmetric bodies and the flow about them require excessively large computer time and a great deal of foresight for the selection of the number and position of the singularities and control points so as to achieve satisfactory results. Furthermore, they provide no insight as to how the errors (e.g., normal velocity) may be progressively reduced.

2. Through the use of the method of least squares and the Automated Design Synthesis optimization, together with a sufficient number of discrete singularities and control points, one can represent an axisymmetric body and the flow about it with excellent accuracy. The bodies are not required to have fore and aft symmetry.

3. Extensive calculations with various types of axisymmetric bodies have shown that the use of two singularities per unit length is quite adequate to model the flow about the body.

4. Numerous examples have been given and the results have been compared with those obtained theoretically.

VI. RECOMMENDATIONS

The following recommendations are made for the purpose of increasing the power of prediction of the dual optimization methodology.

1. The existing computer program should be converted to an interactive mode so as to simplify its use.

2. The code should be enhanced to enable the user to minimize not only the strength and location of the singularities but also their number. For example, such a procedure could be incorporated into the code by deleting from the calculations the singularities whose strengths fall below a prescribed percentage of the sum of the absolute values of the strengths of all singularities.

3. The improved code should be tested with other axisymmetric body shapes.

LIST OF REFERENCES

1. von Karman, Th., "Calculation of the Flowfield Around Airships," NACA TM 574, July 1930.

2. Hess J. L., "Review of Integral-Equation Techniques for Solving Potential Flow Problems with Emphasis on the Surface-Source Method," Computational Methods in Applied Mechanics and Engineering, Vol. 5, 1975, p. 145.

3. Oberkampf, W. L. and Watson, L. E., "Incompressible Potential Flow Solutions for Arbitrary Bodies of Revolution," AIAA Journal, Vol. 12, March 1974, pp. 409-411.

4. Karamcheti, K., Principles of Ideal-Fluid Aerodynamics, Wiley & Sons, New York, 1966.

5. Milne-Thomson, L. M., Theoretical Hydrodynamics, (4th Ed.), The MacMillan Co., New York 1960, (pp. 475-479).

6. Sheldon, W., Kolansky, M.S., Gluckman, M.T., and Pfeffer, R., "An Approximate Theory of Incompressible Viscous Flow Past Two-Dimensional Bluff Bodies in the Intermediate Reynolds Number Regime," Journal of Fluid Mechanics, Vol. 77, part 1, 1976, pp. 129-152.

7. Kreyszig, E., Advanced Engineering Mathematics, (3rd Ed.). John Wiley & Sons, Inc., New York, 1972 (pp. 681-683).

8. Vanderplaats, G. N., "ADS - A Fortran Program for Automated Design Synthesis," Version 0.9, Naval Postgraduate School, Monterey, CA, 1983. (unpublished handout for course ME 4731)

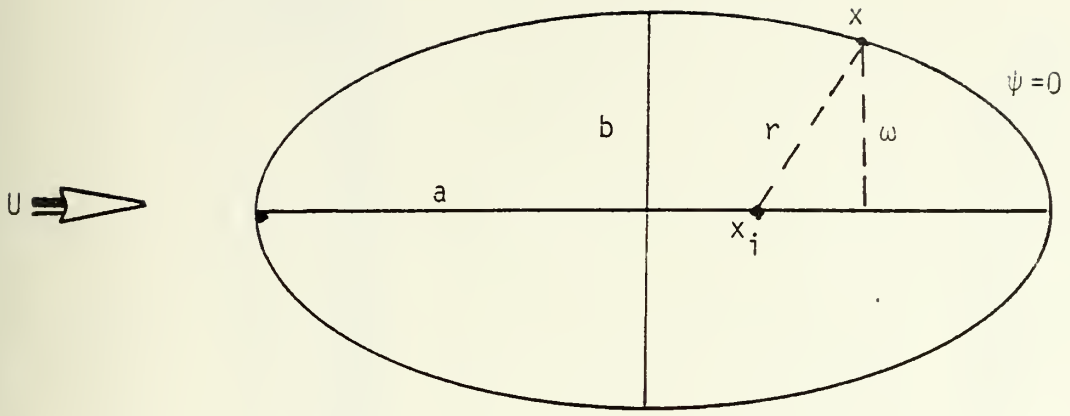


Figure 1. Ovary Ellipsoid

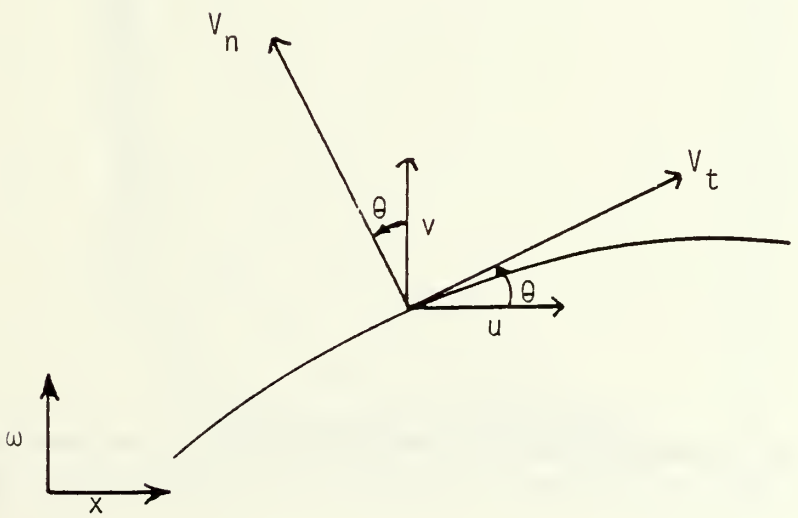


Figure 2. Velocity Components

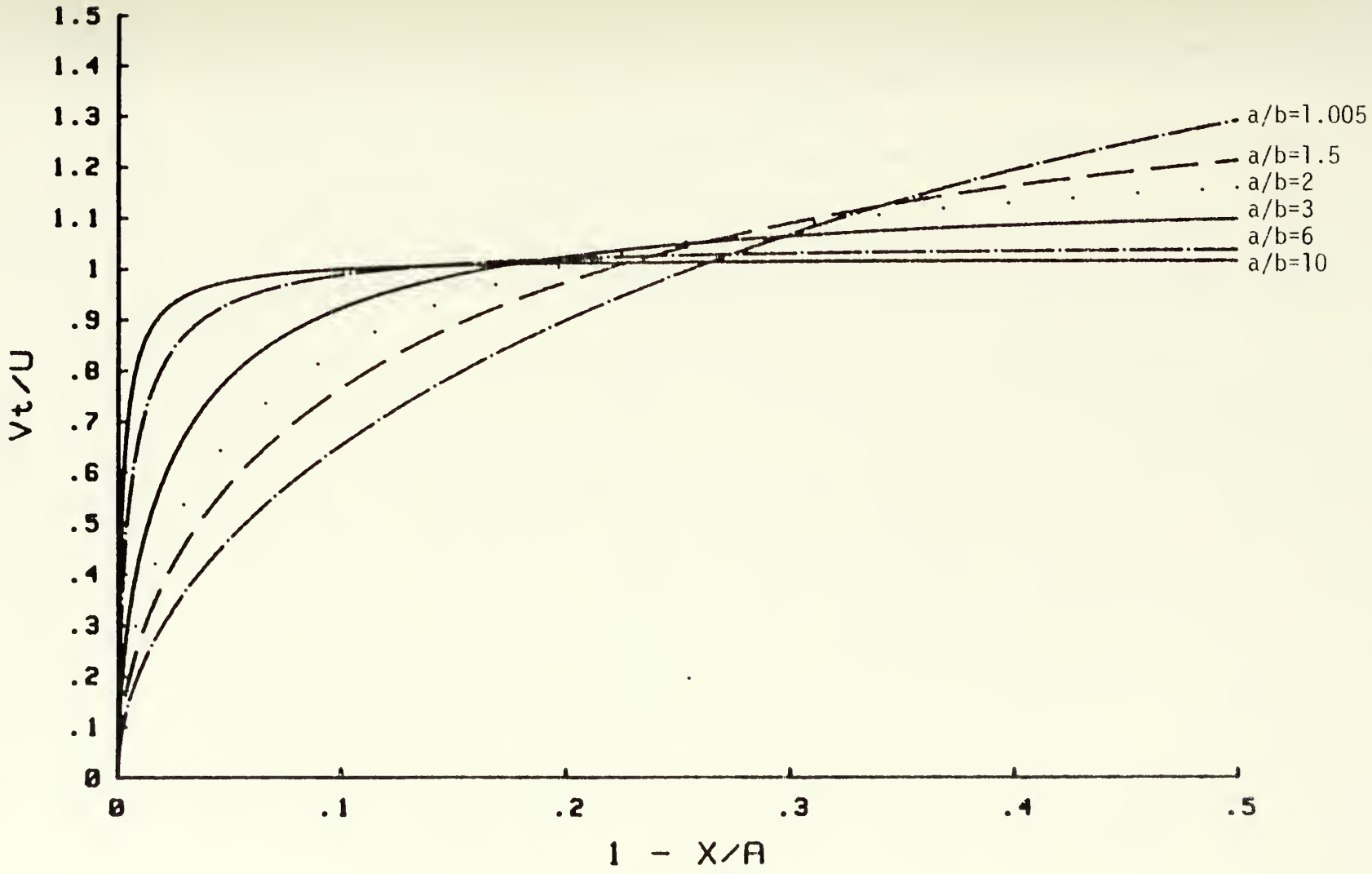


Figure 3a. Theoretical Tangential Velocity for Quarter Length of Body

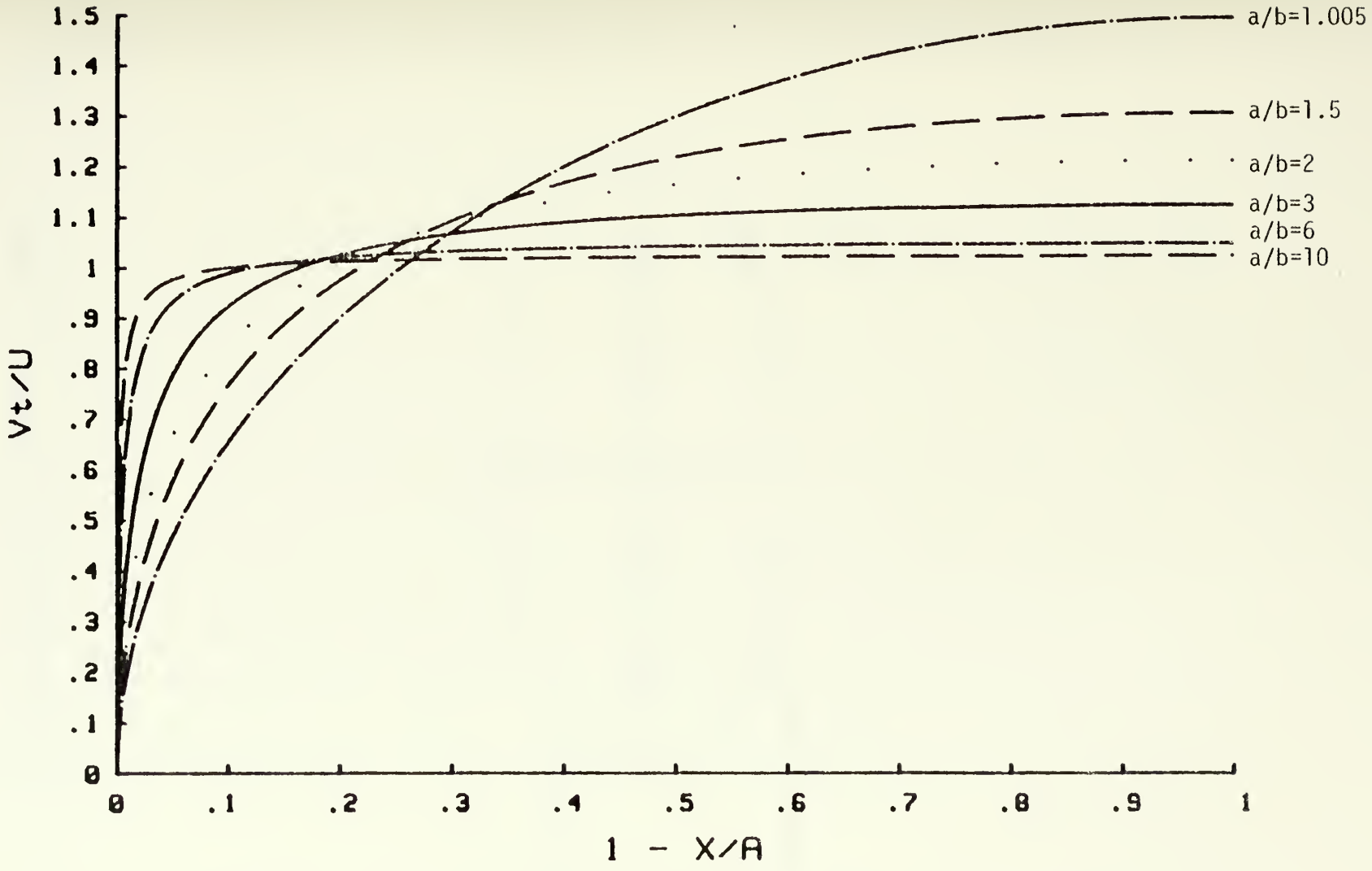


Figure 3b. Theoretical Tangential Velocity for Half Length of Body

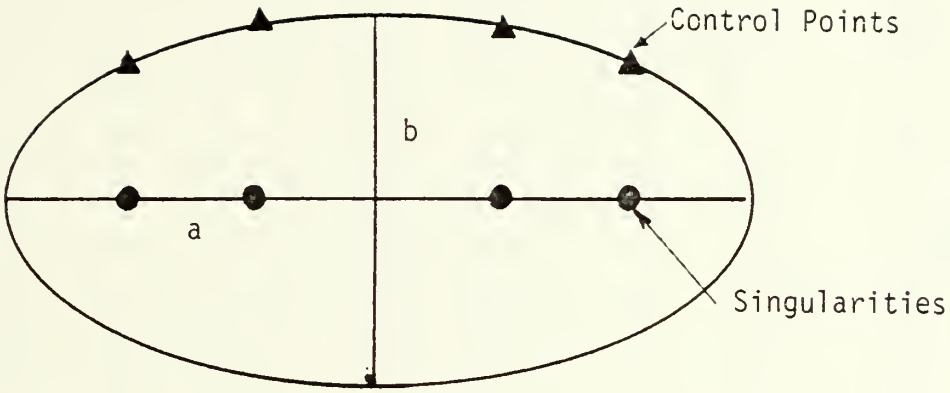


Figure 4. Placement of Singularities and Control Points

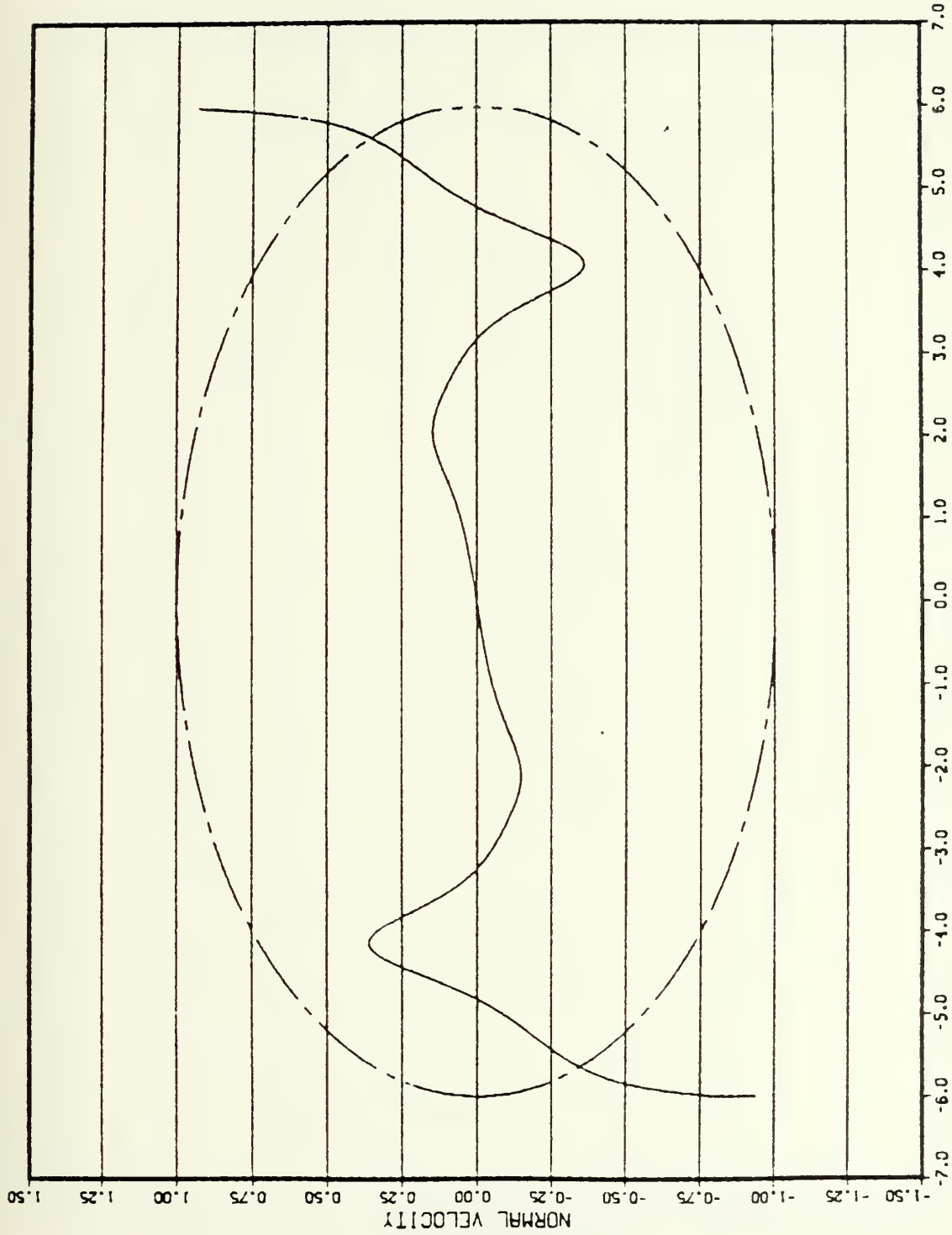


Figure 5a. Normal Velocity for the Case of 4/4

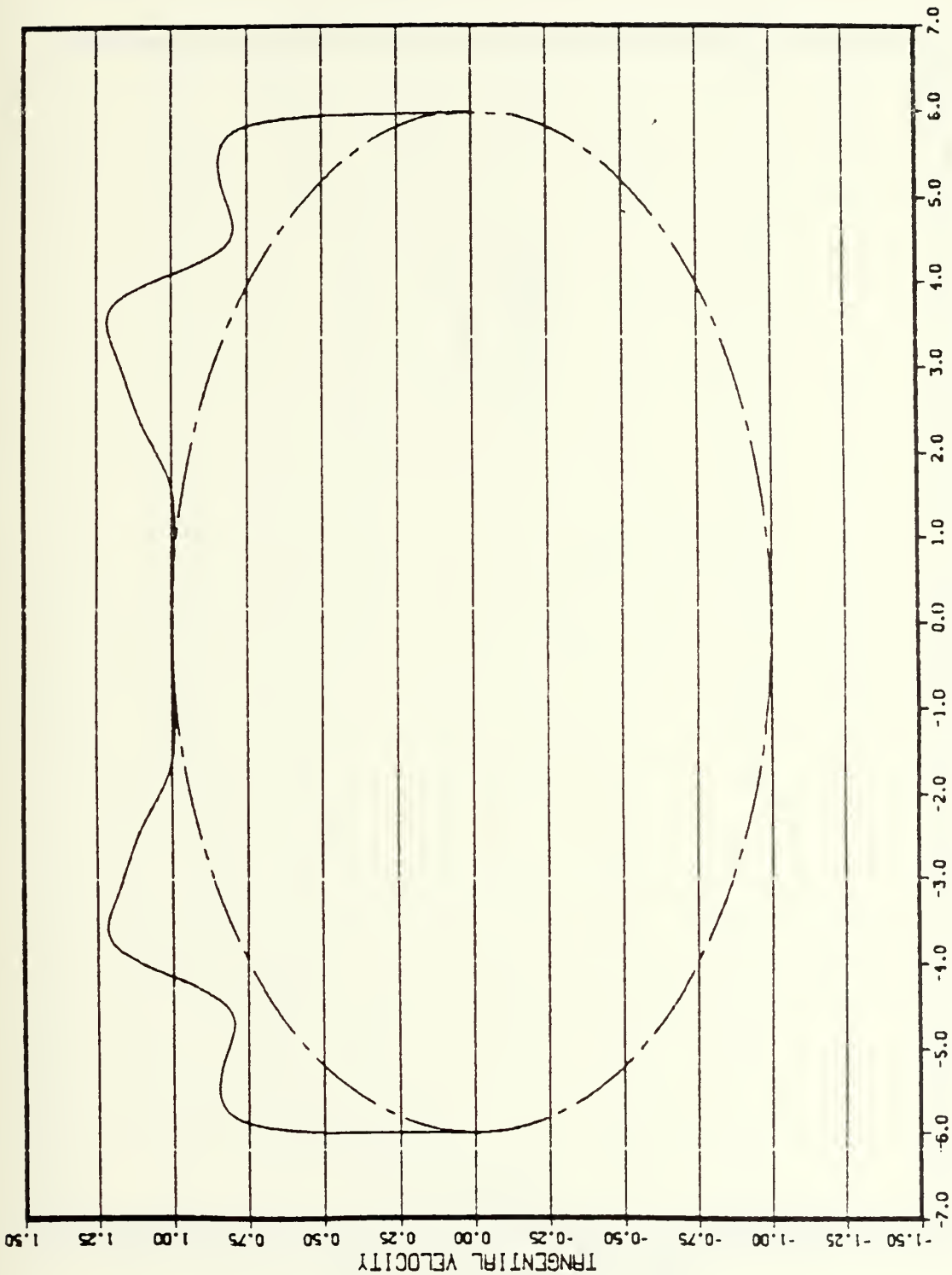


Figure 5b. Tangential Velocity for the Case of 4/4

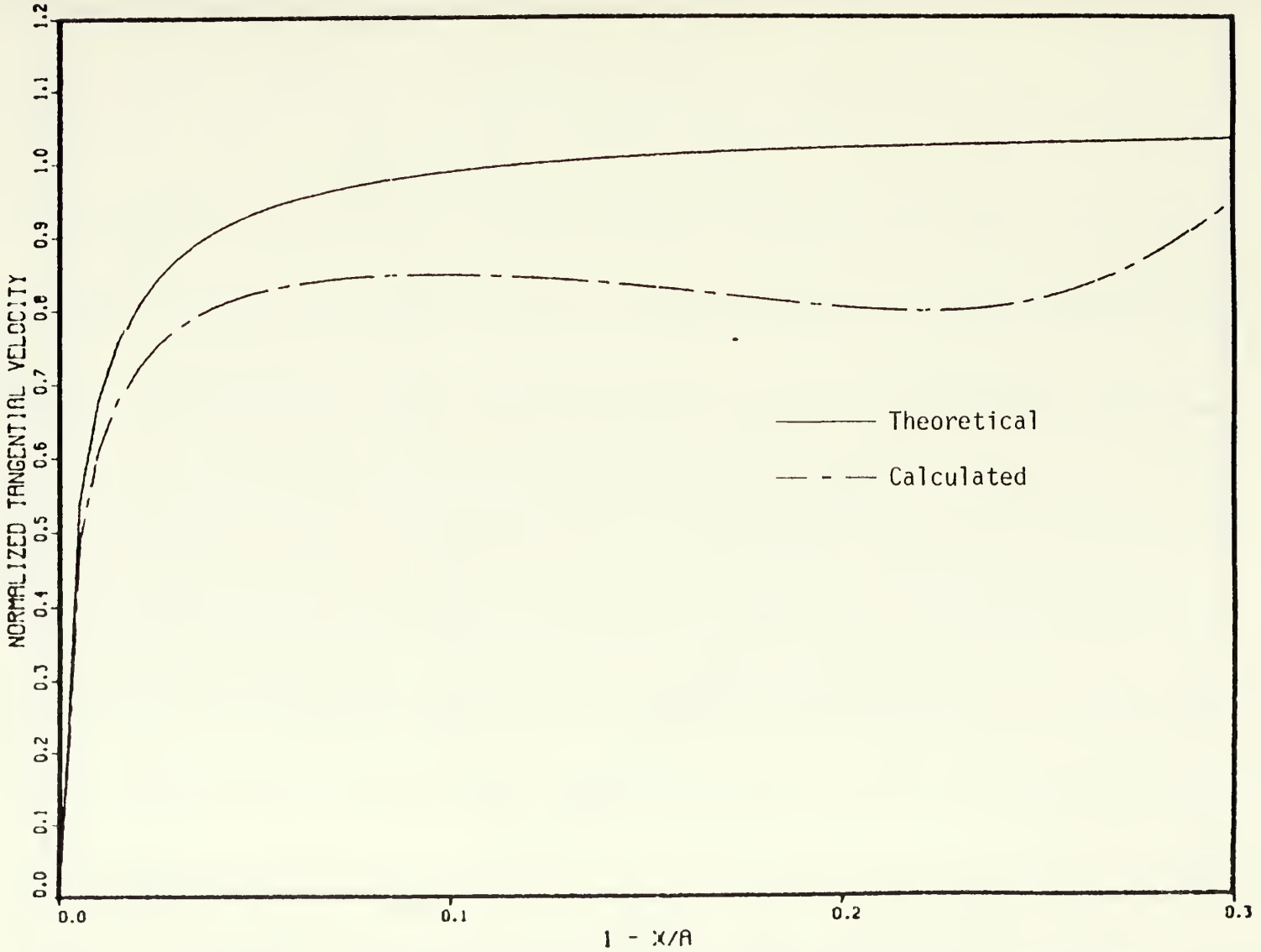


Figure 5c. Theoretical and Calculated Tangential Velocities for the Case 4/4

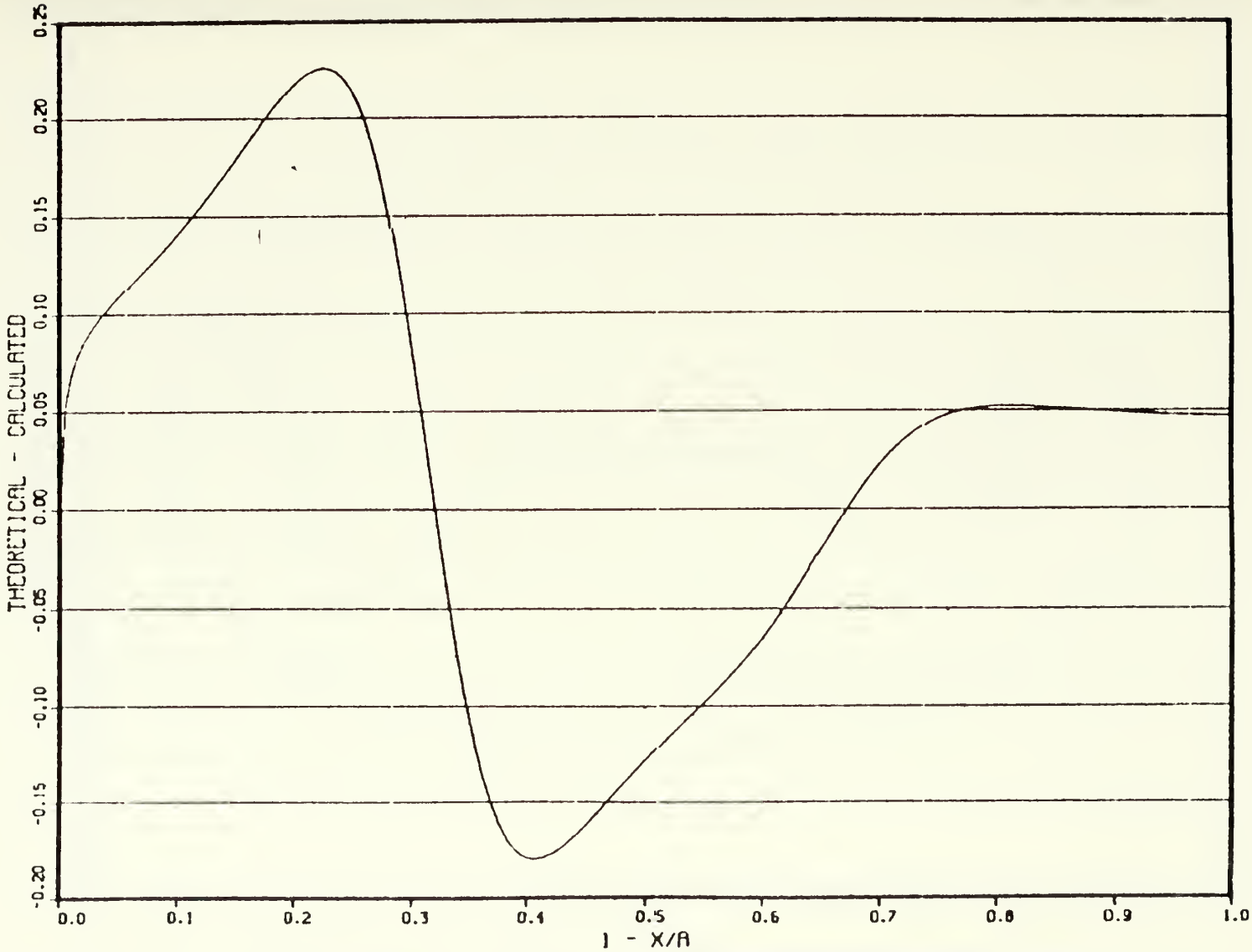


Figure 5d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 4/4

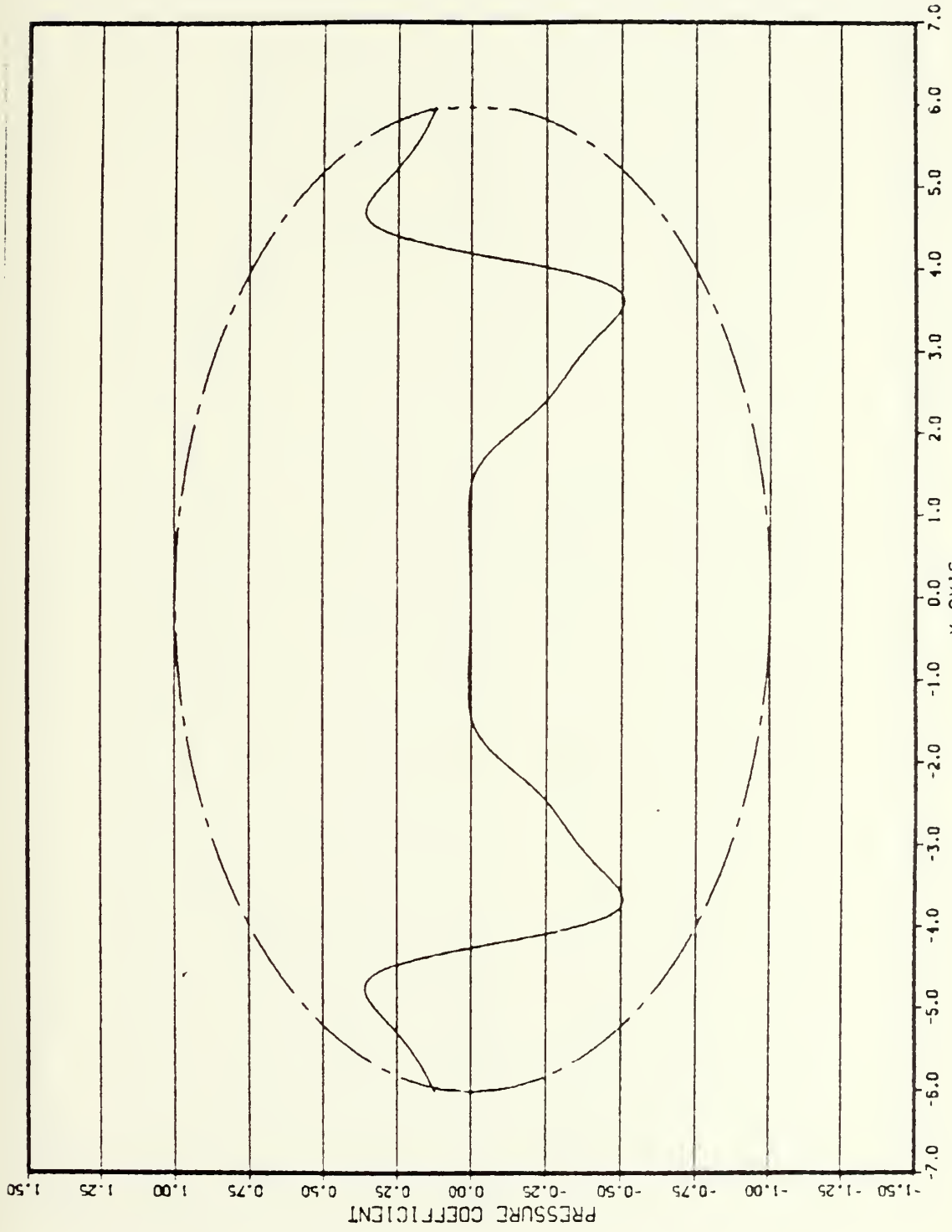


Figure 5e. Pressure Coefficient for the Case of 4/4

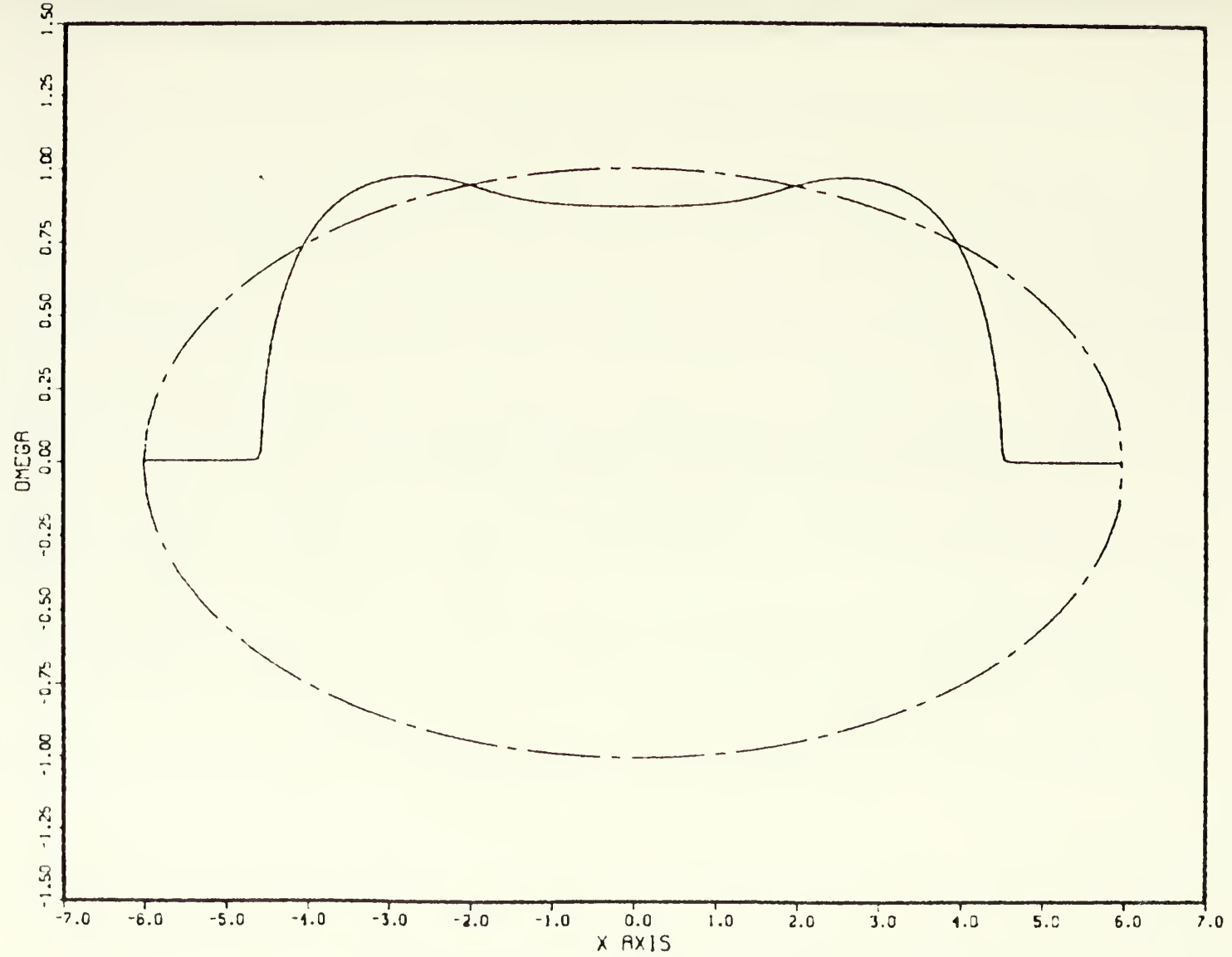


Figure 5f. Body Shape for the Case of 4/4

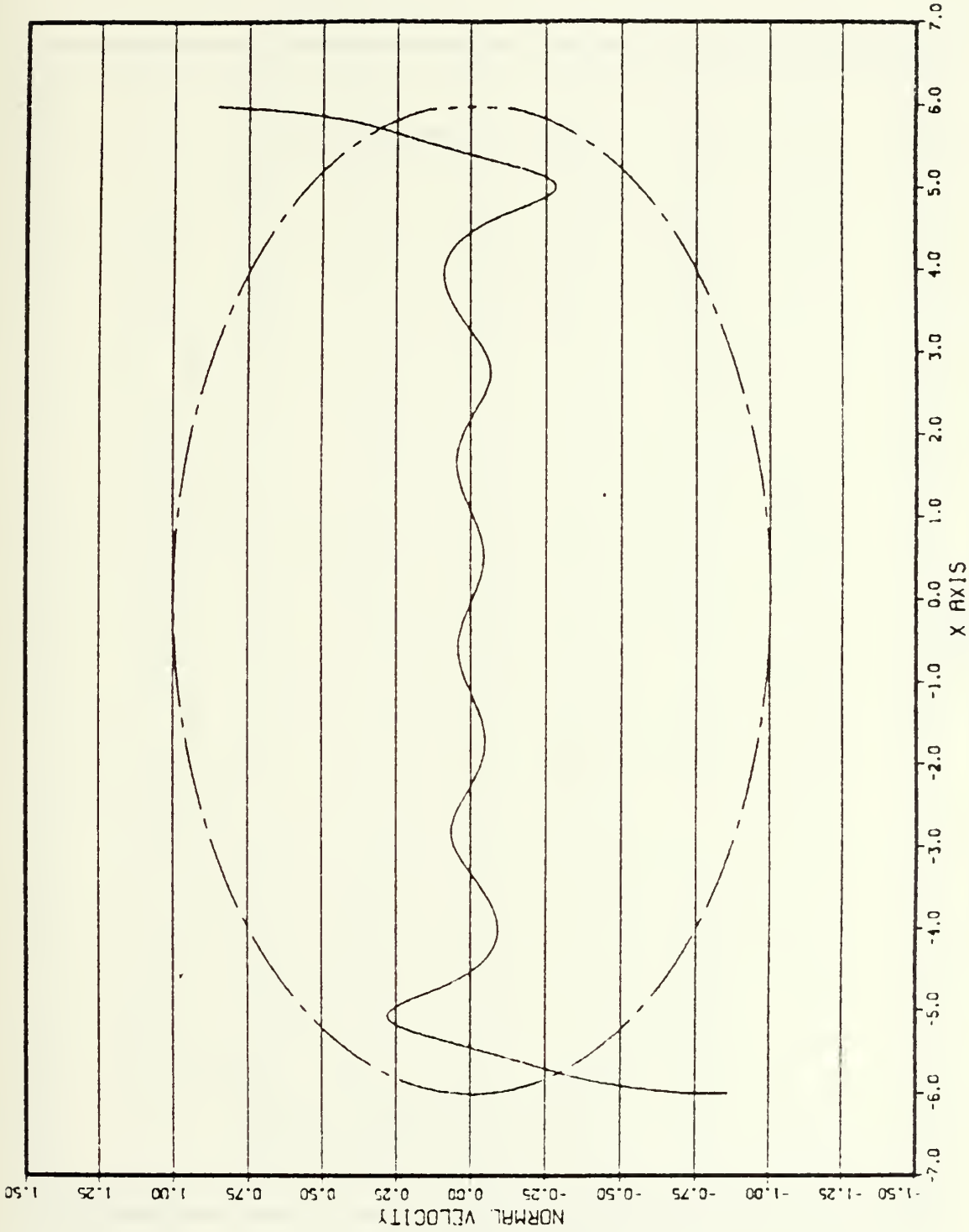


Figure 6a. Normal Velocity for the Case of 10/10

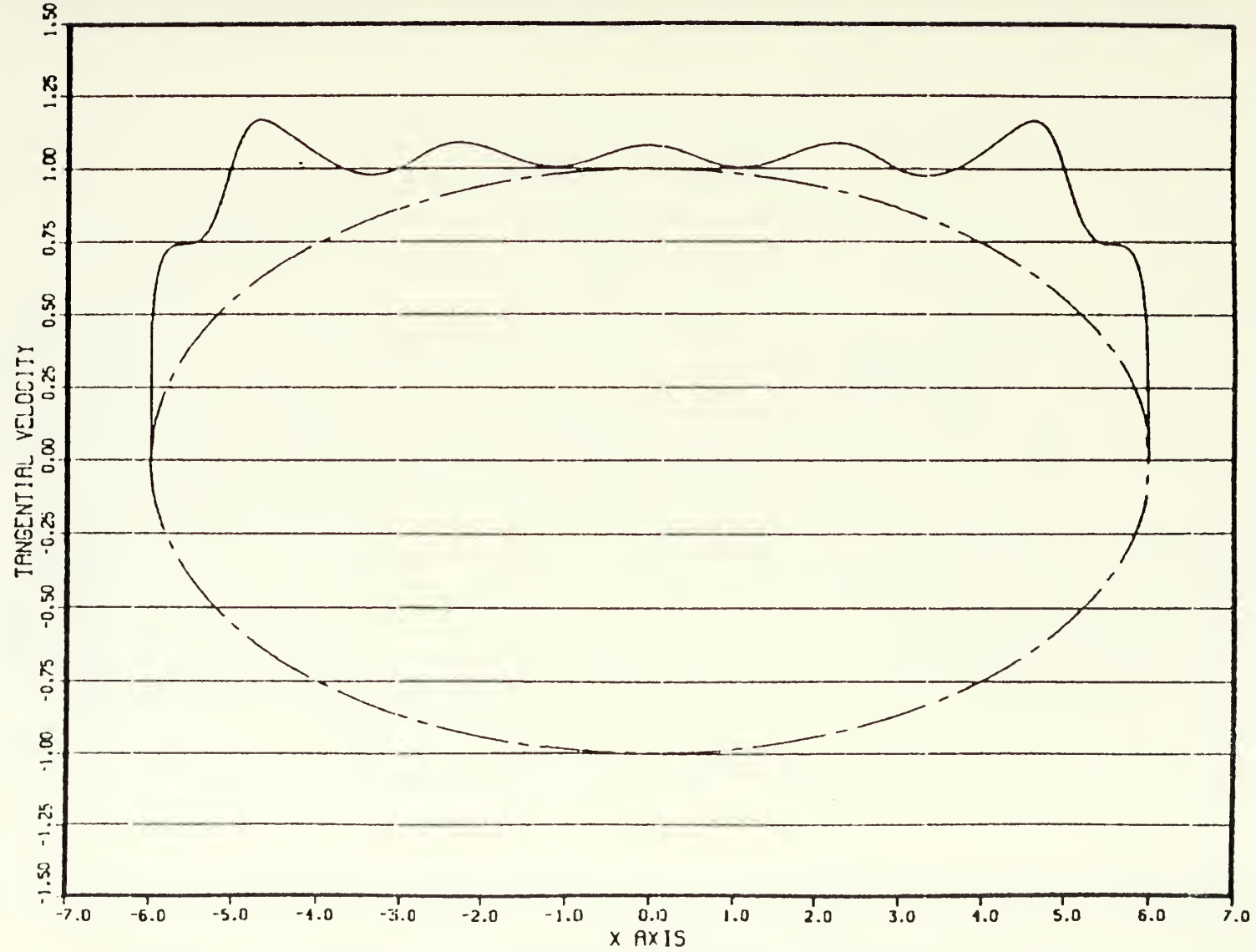


Figure 6b. Tangential Velocity for the Case of 10/10

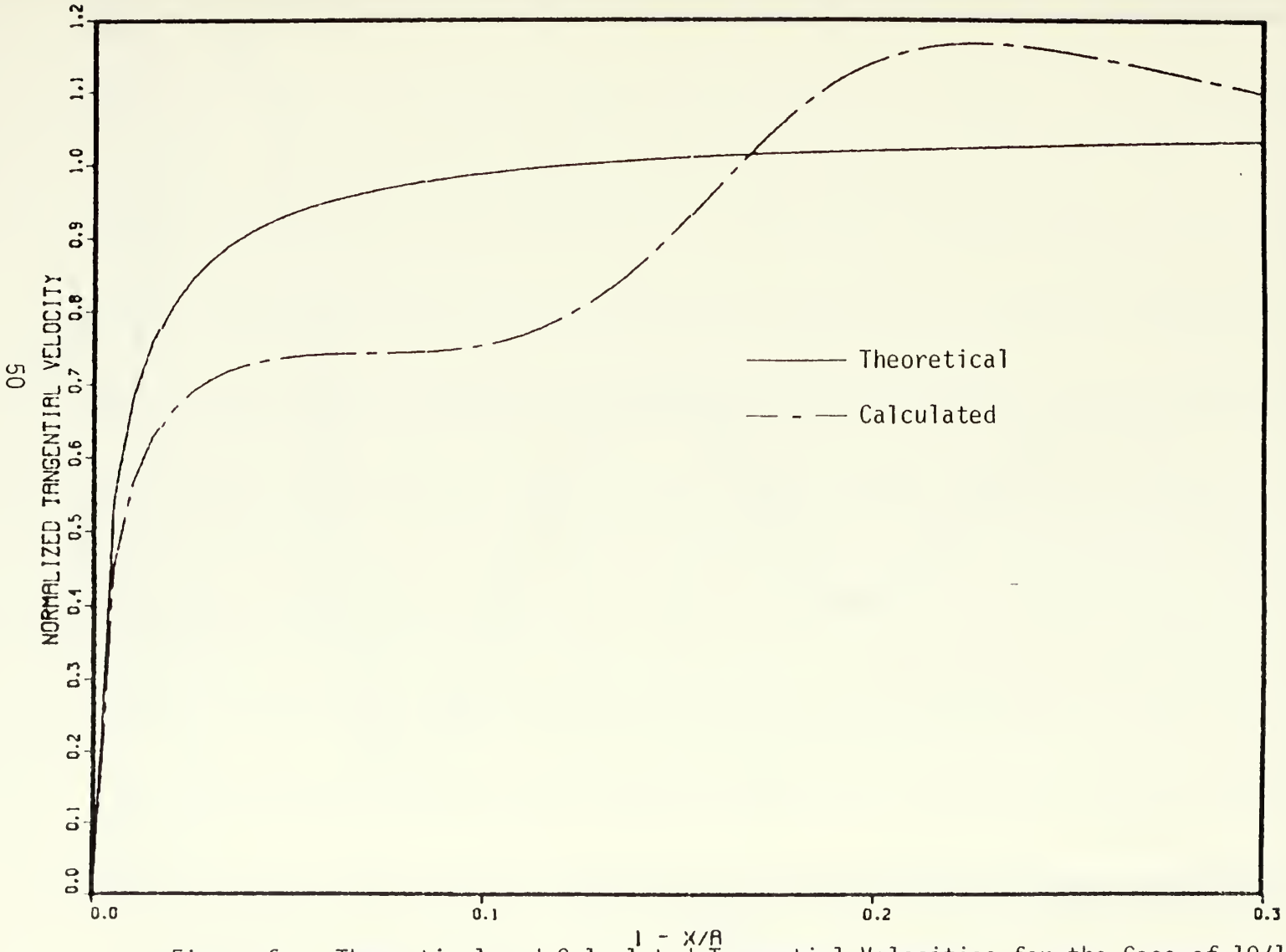


Figure 6c. Theoretical and Calculated Tangential Velocities for the Case of 10/10

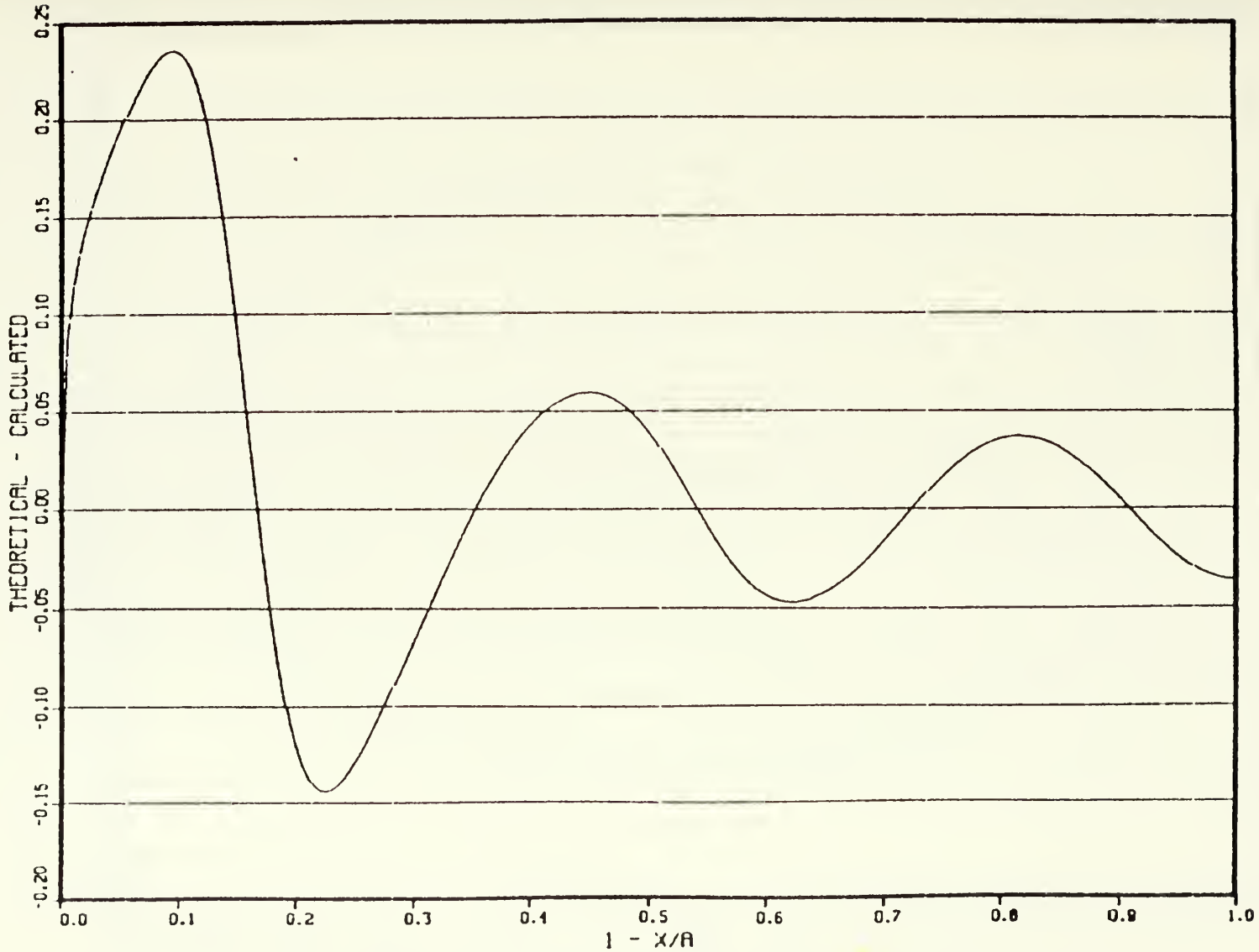


Figure 6d. Difference between the Theoretical and Calculated Tangential Velocities for Case of 10/10

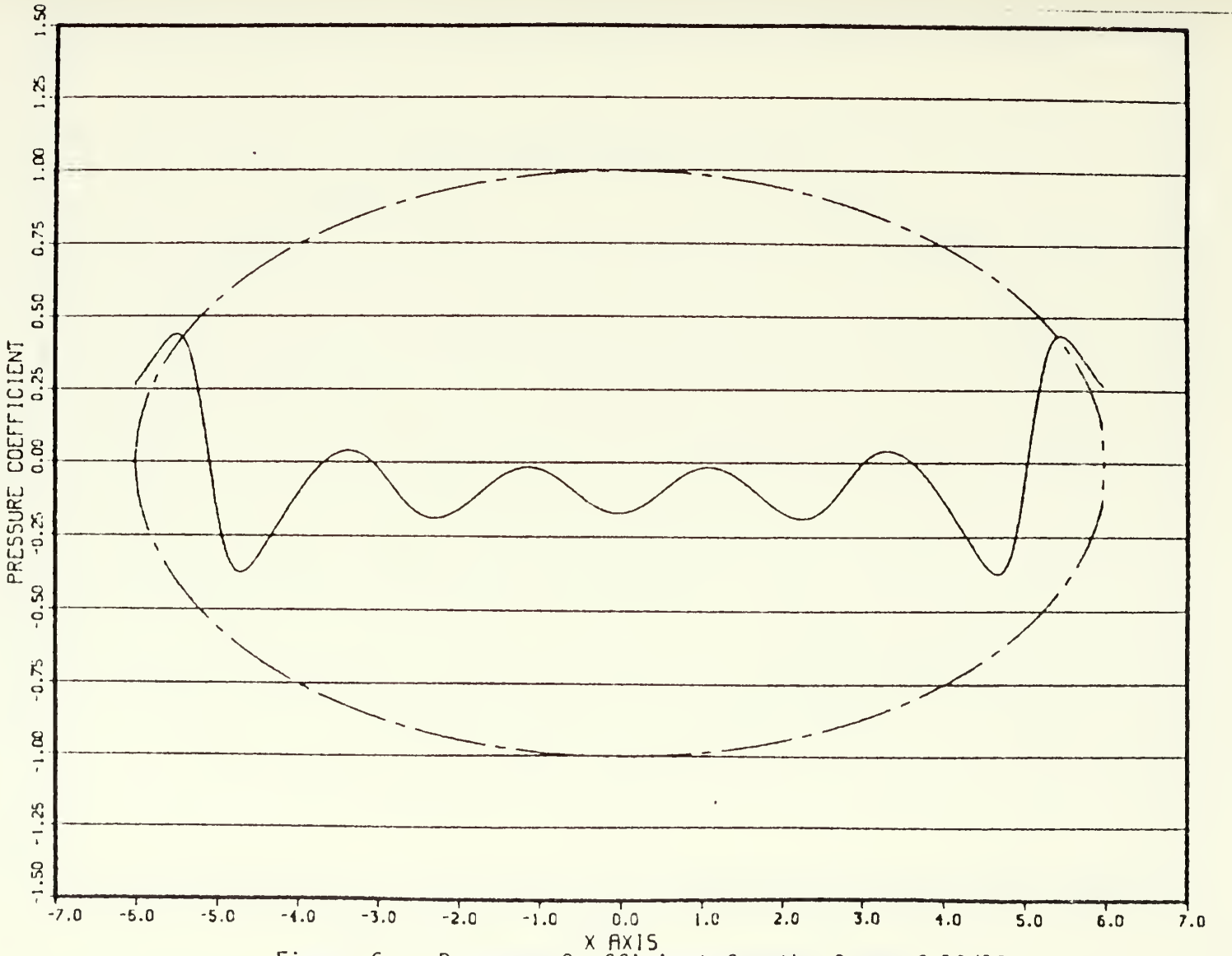


Figure 6e. Pressure Coefficient for the Case of 10/10

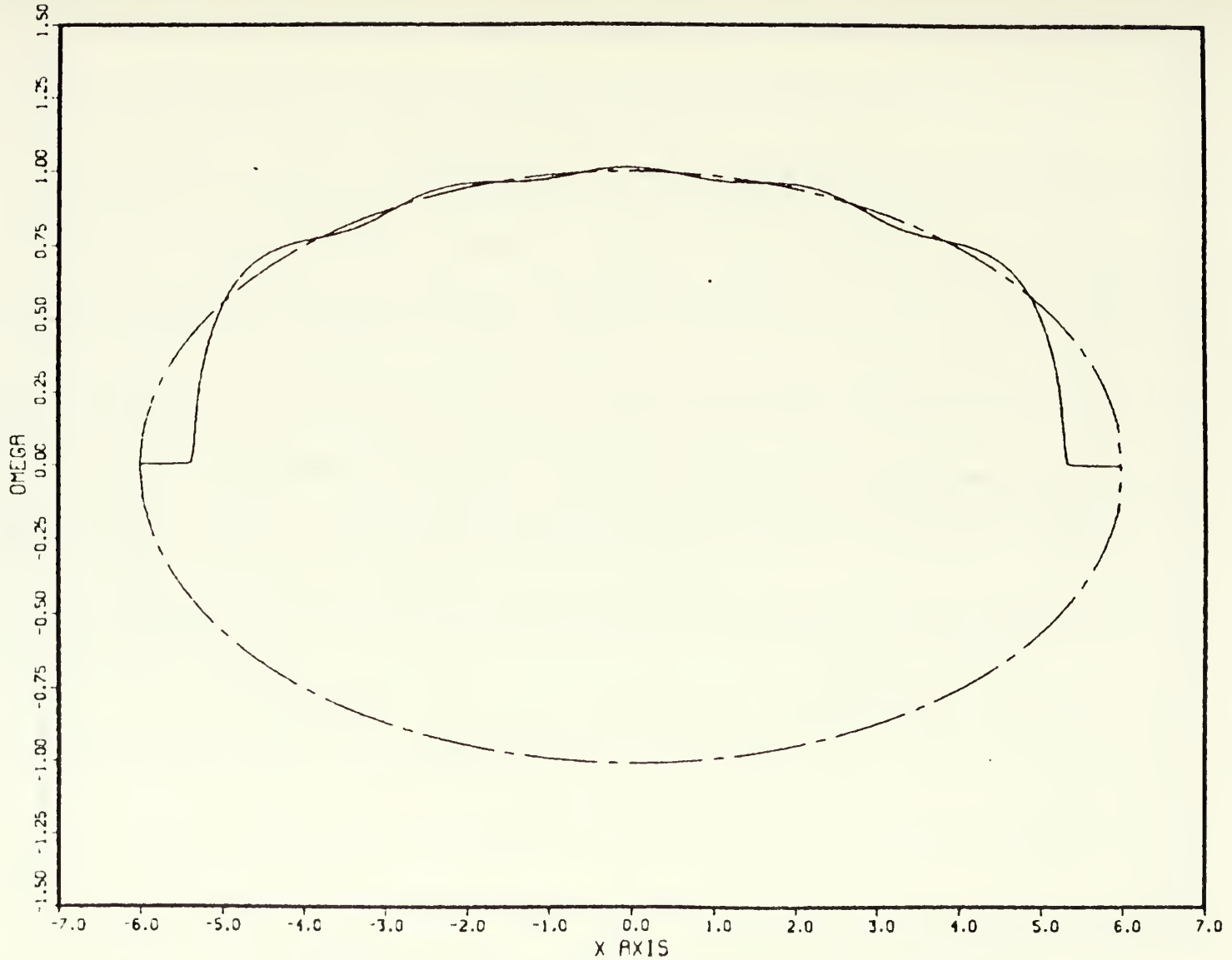


Figure 6f. Body Shape for the Case of 10/10

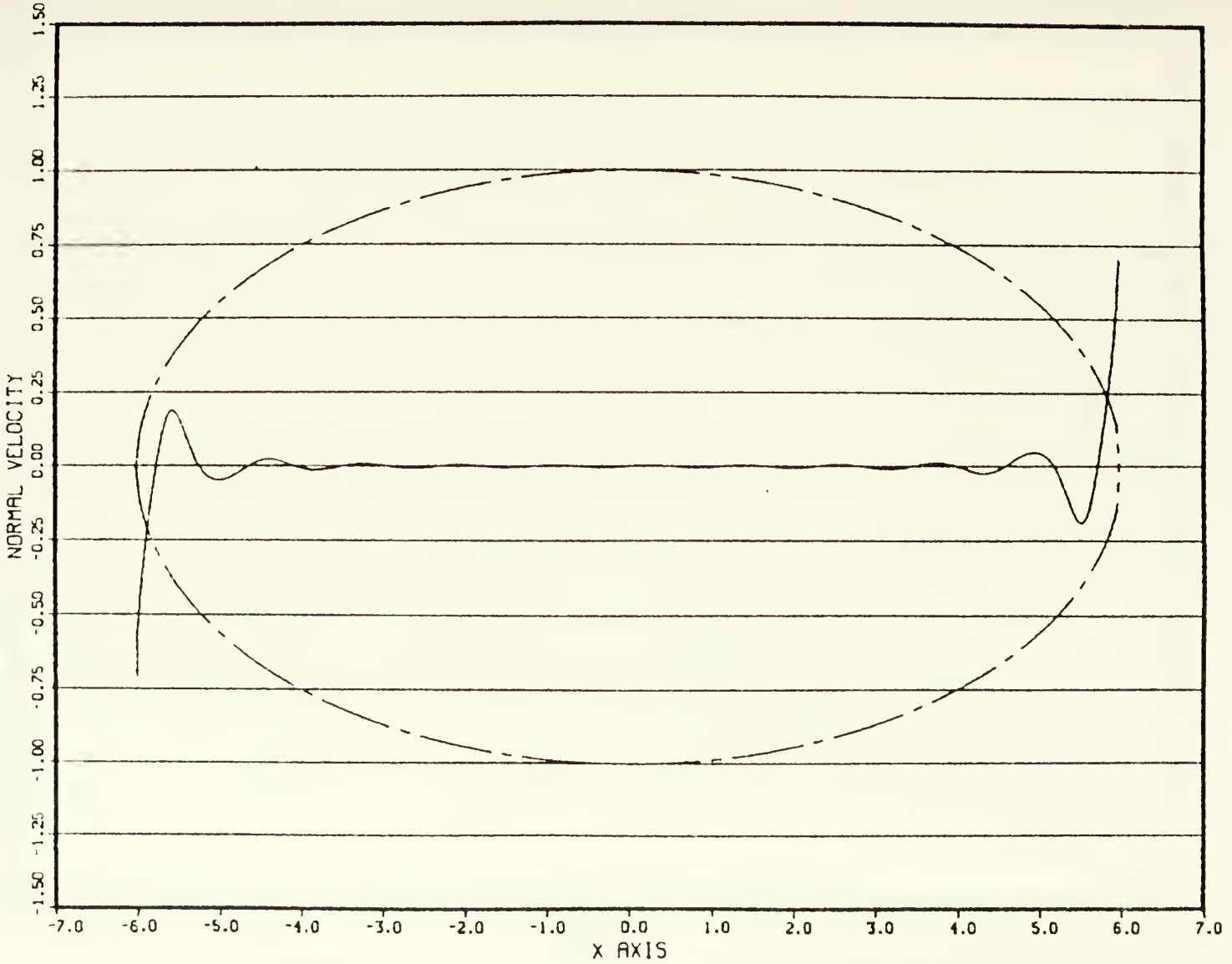


Figure 7a. Normal Velocity for the Case of 20/20

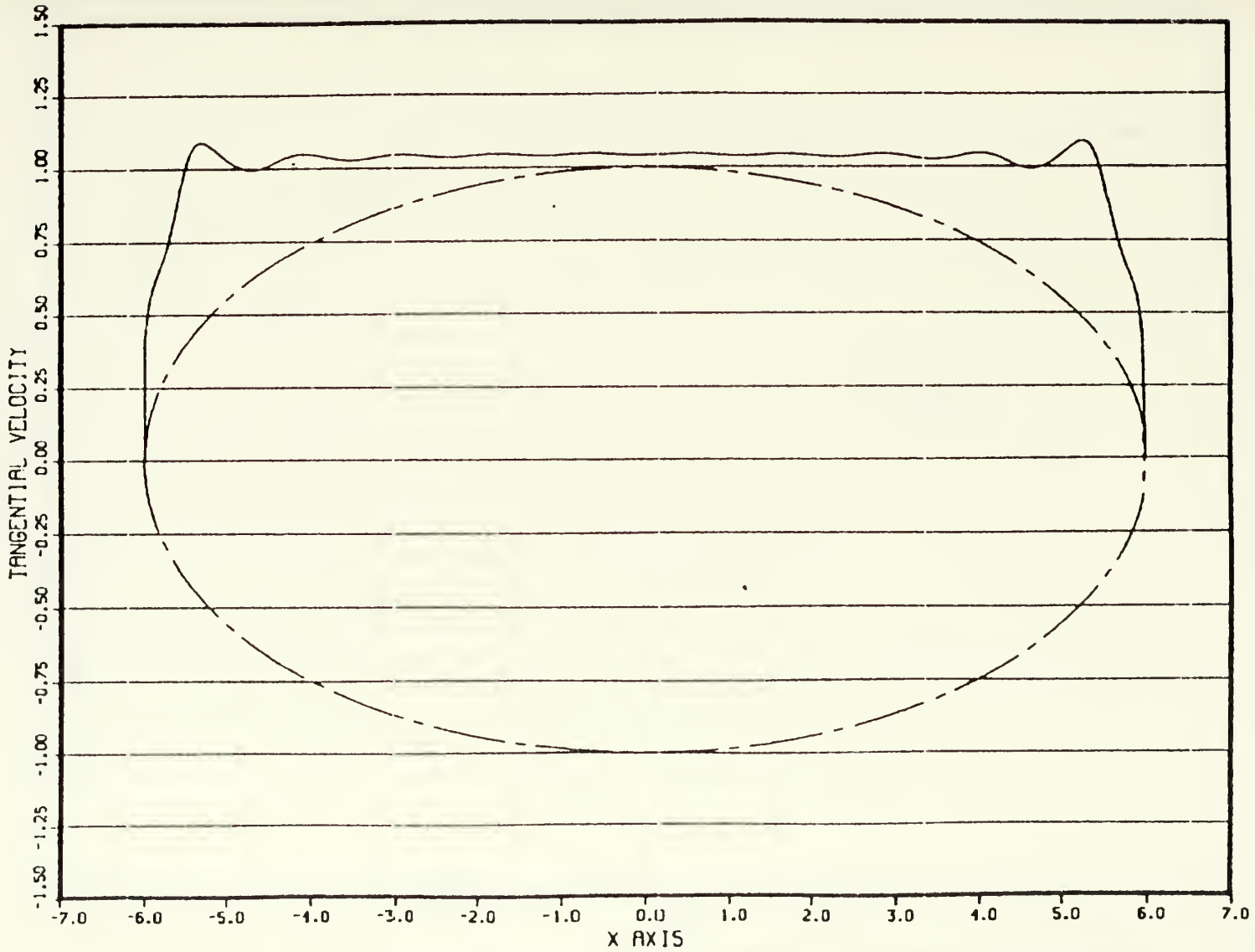


Figure 7b. Tangential Velocity for the Case of 20/20

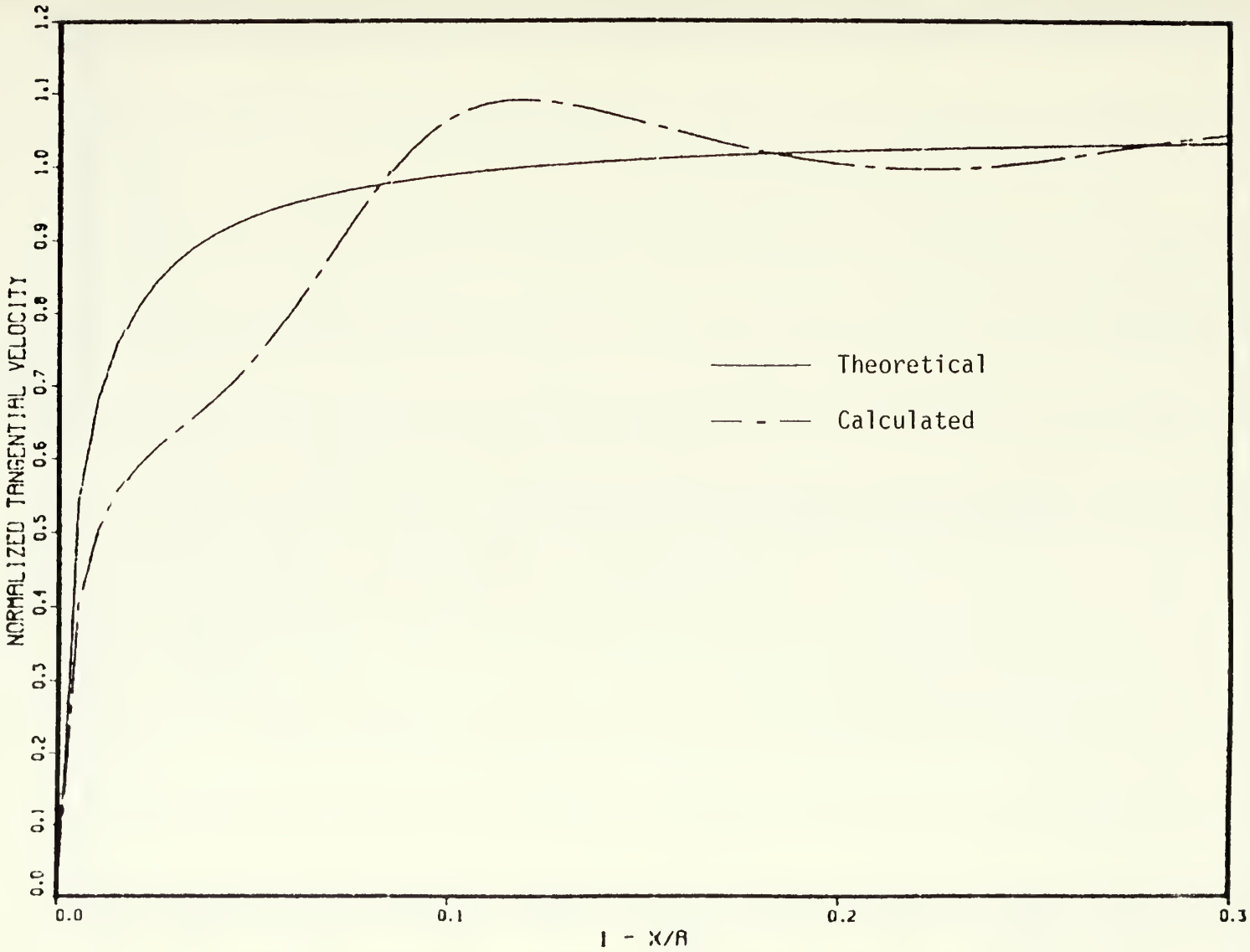


Figure 7c. Theoretical and Calculated Tangential Velocities for the Case of 20/20

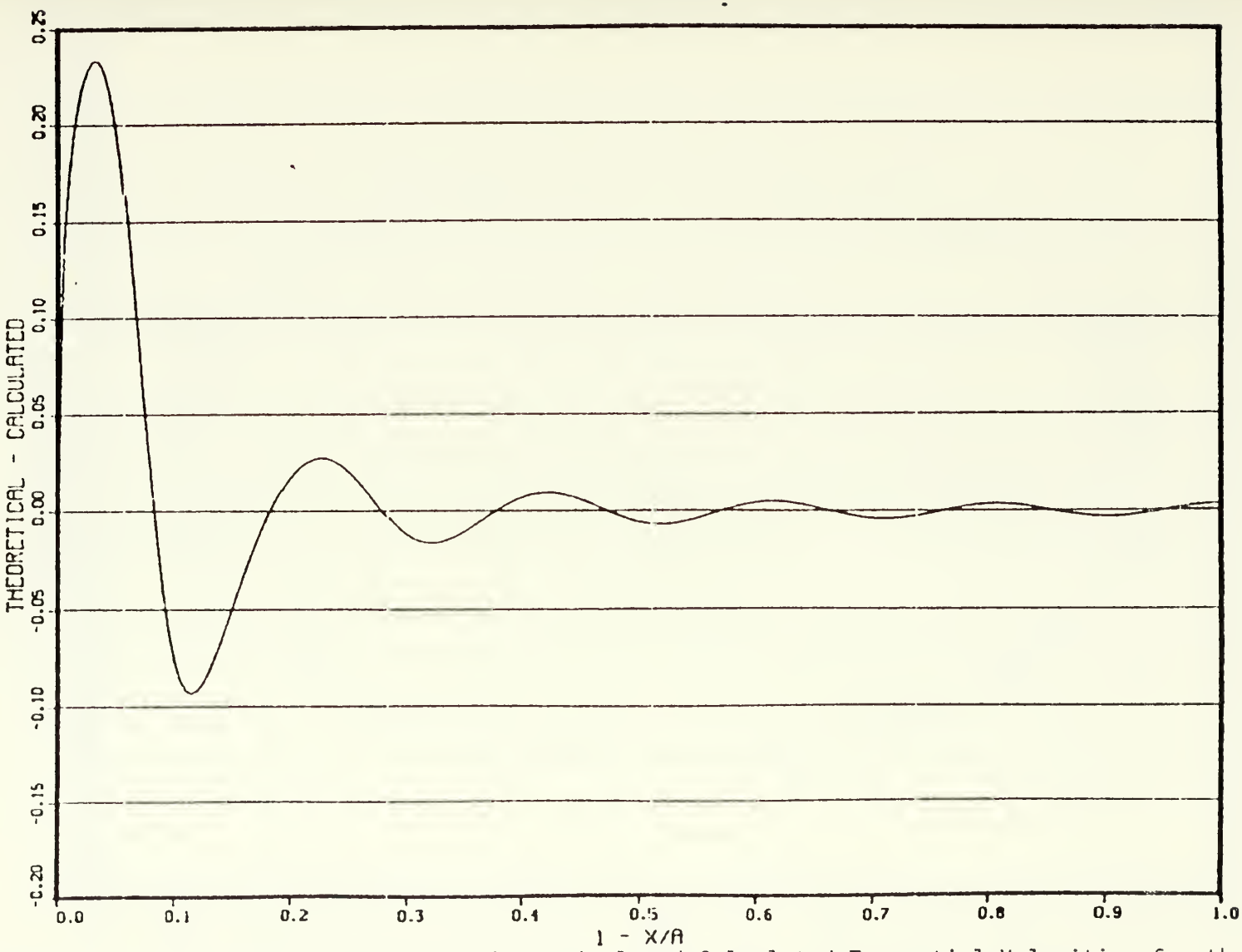


Figure 7d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 20/20

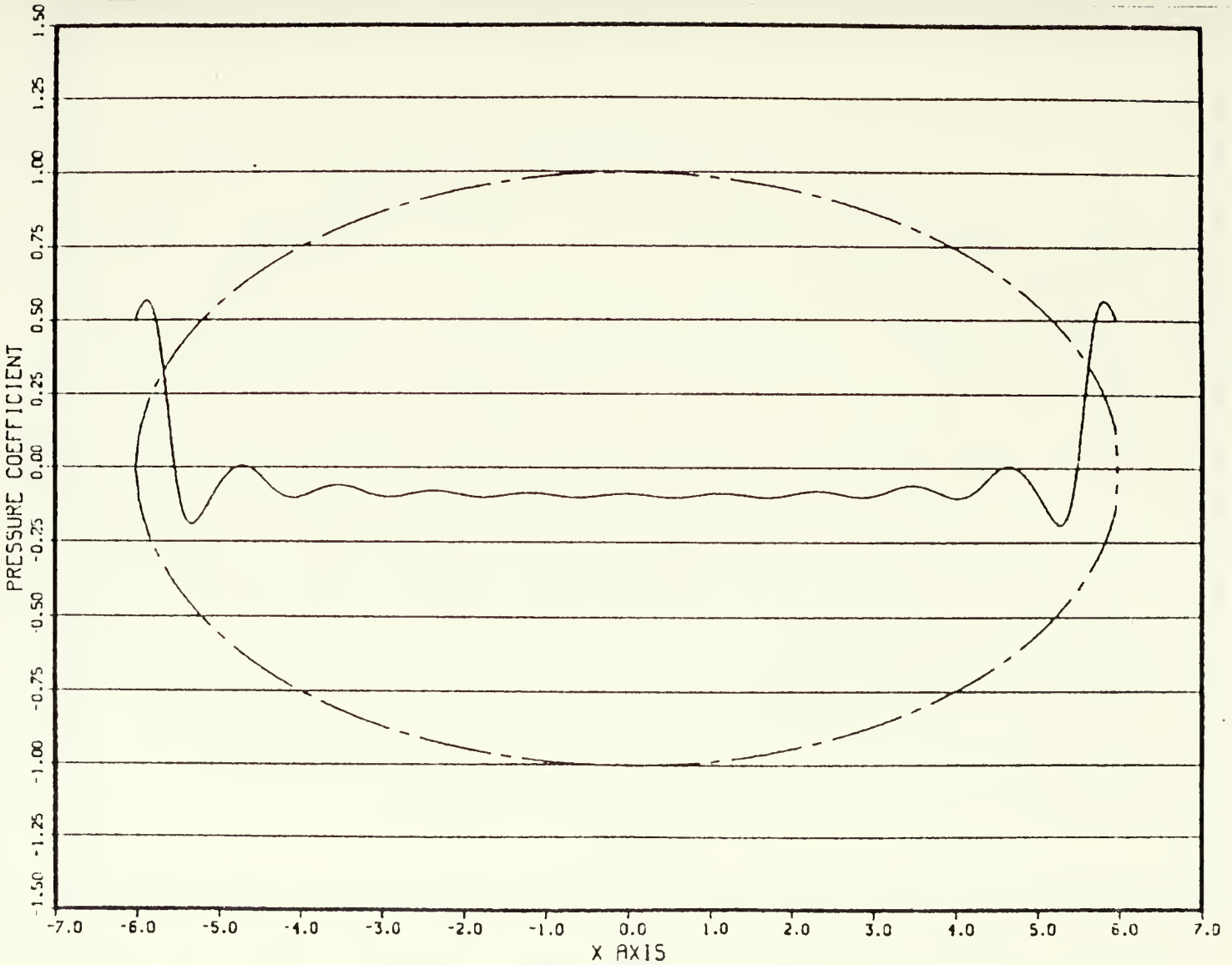


Figure 7e. Pressure Coefficient for the Case of 20/20

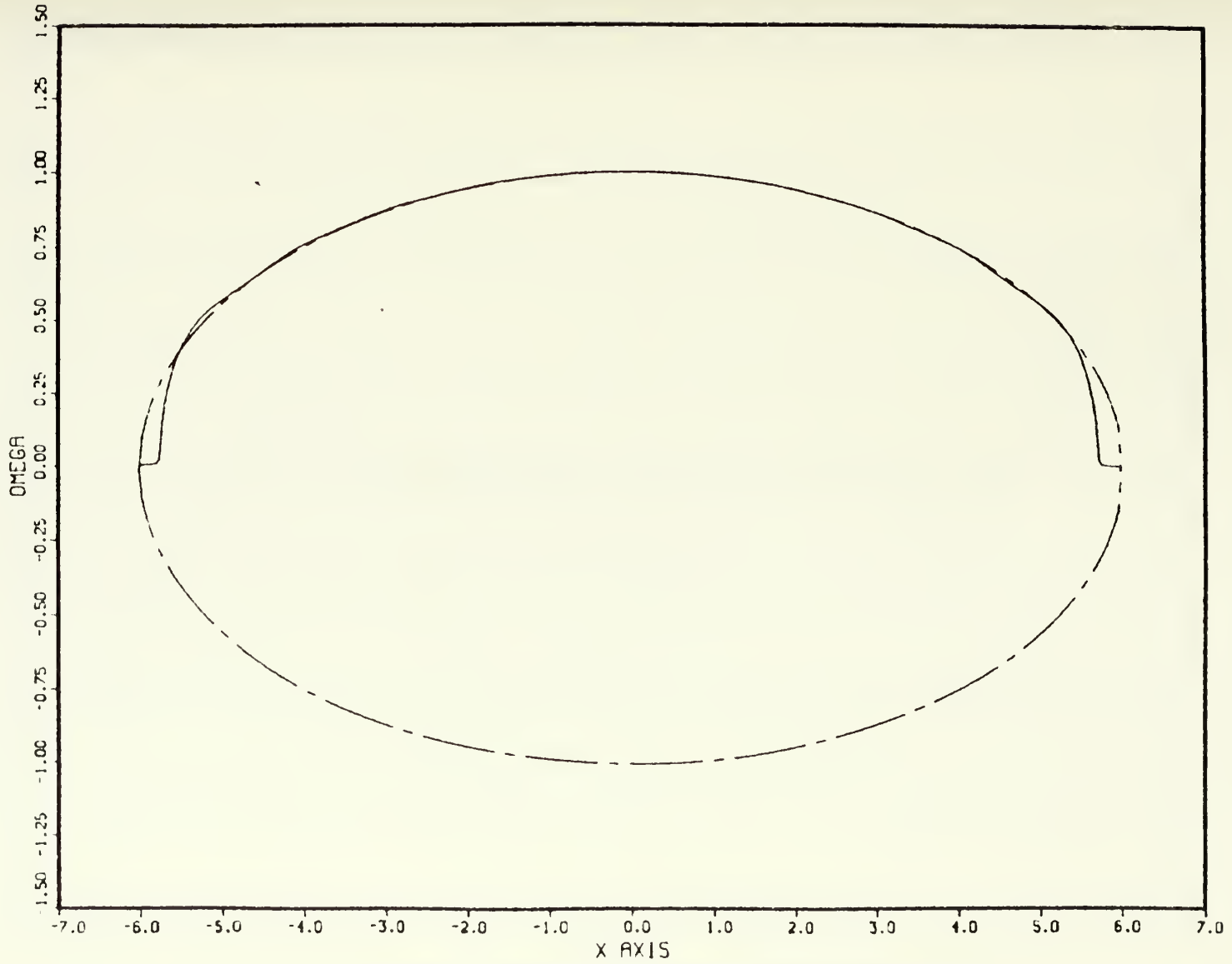


Figure 7f. Body Shape for the Case of 20/20

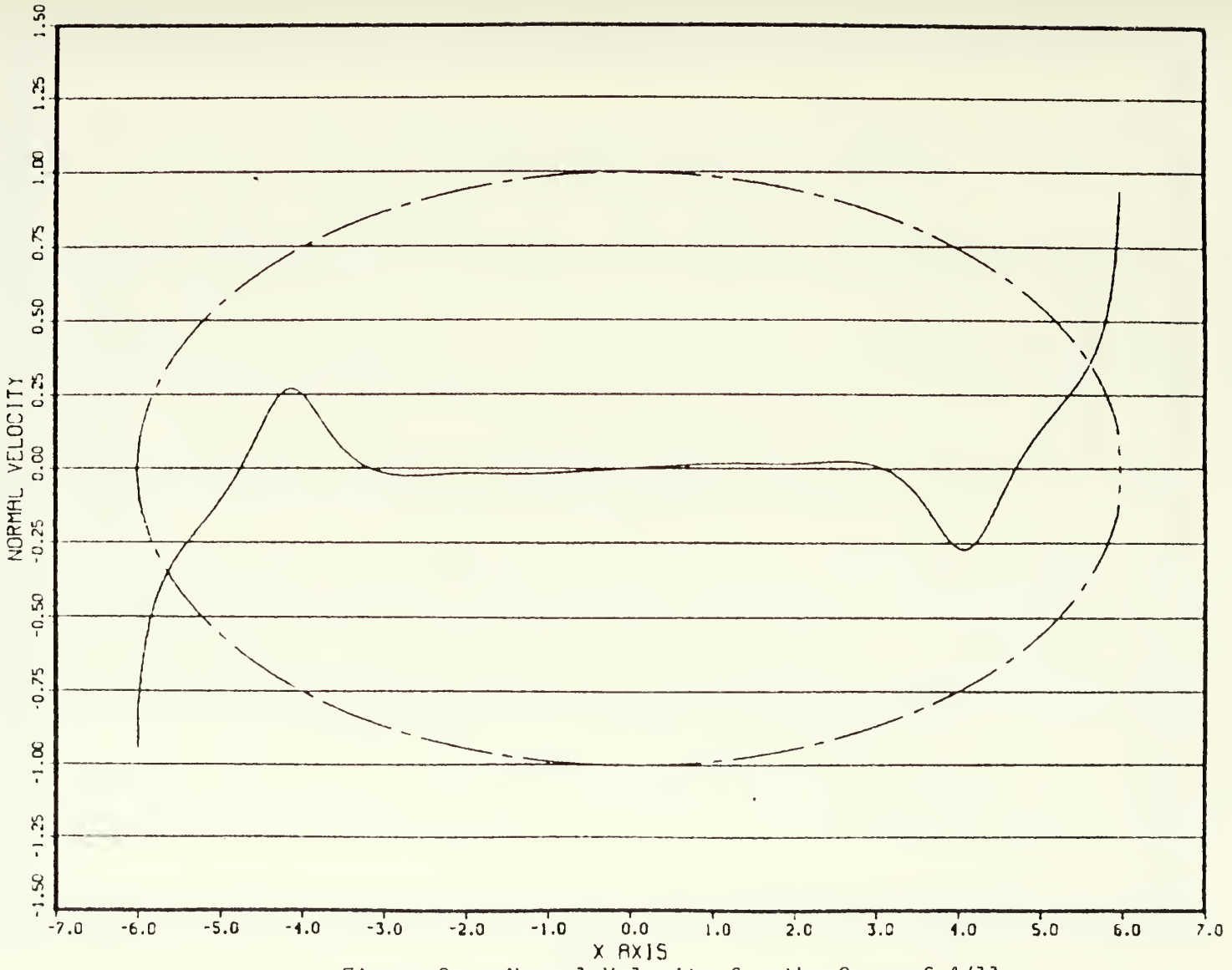


Figure 8a. Normal Velocity for the Case of 4/11

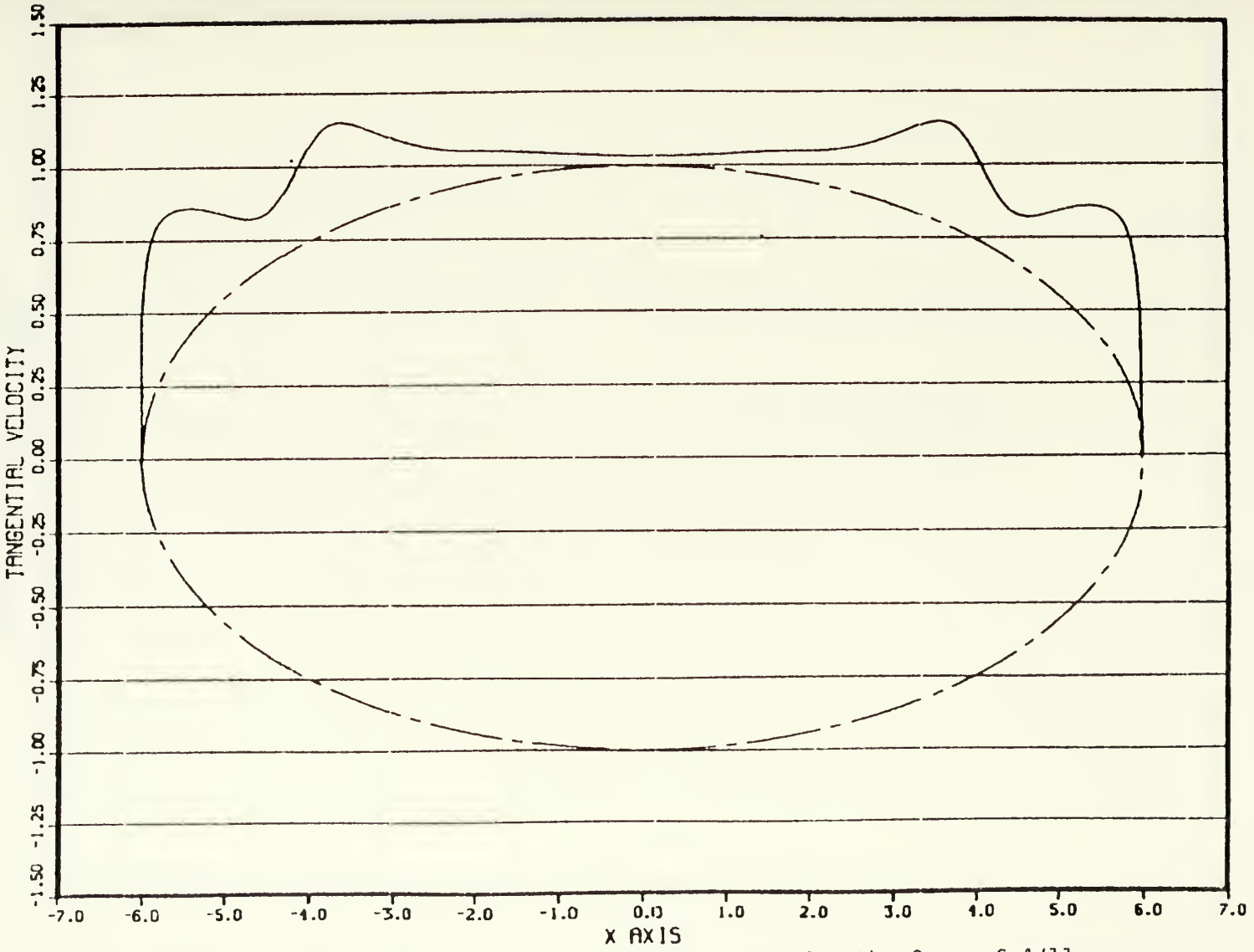


Figure 8b. Tangential Velocity for the Case of 4/11

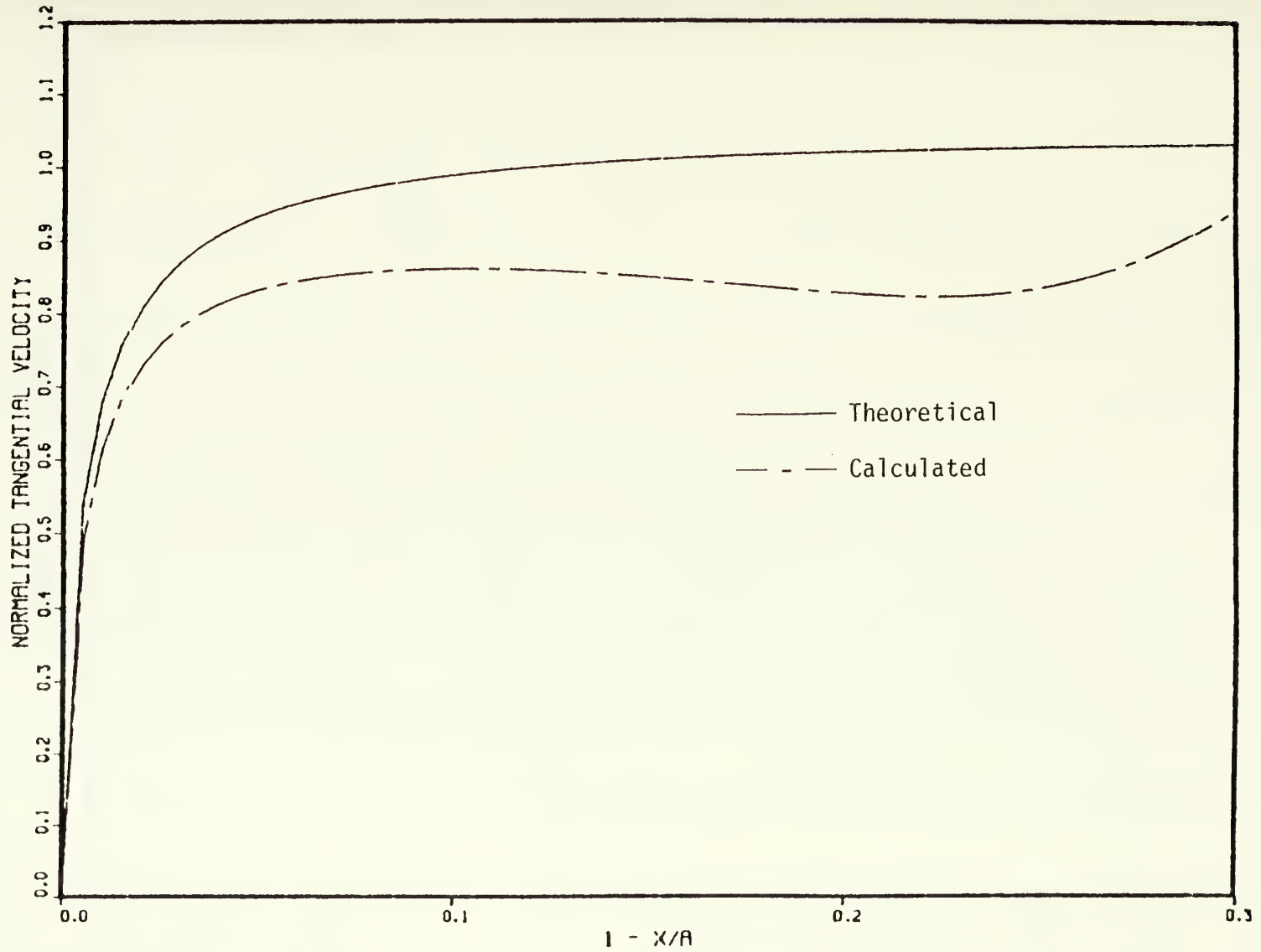


Figure 8c. Theoretical and Calculated Tangential Velocities for the Case of 4/11

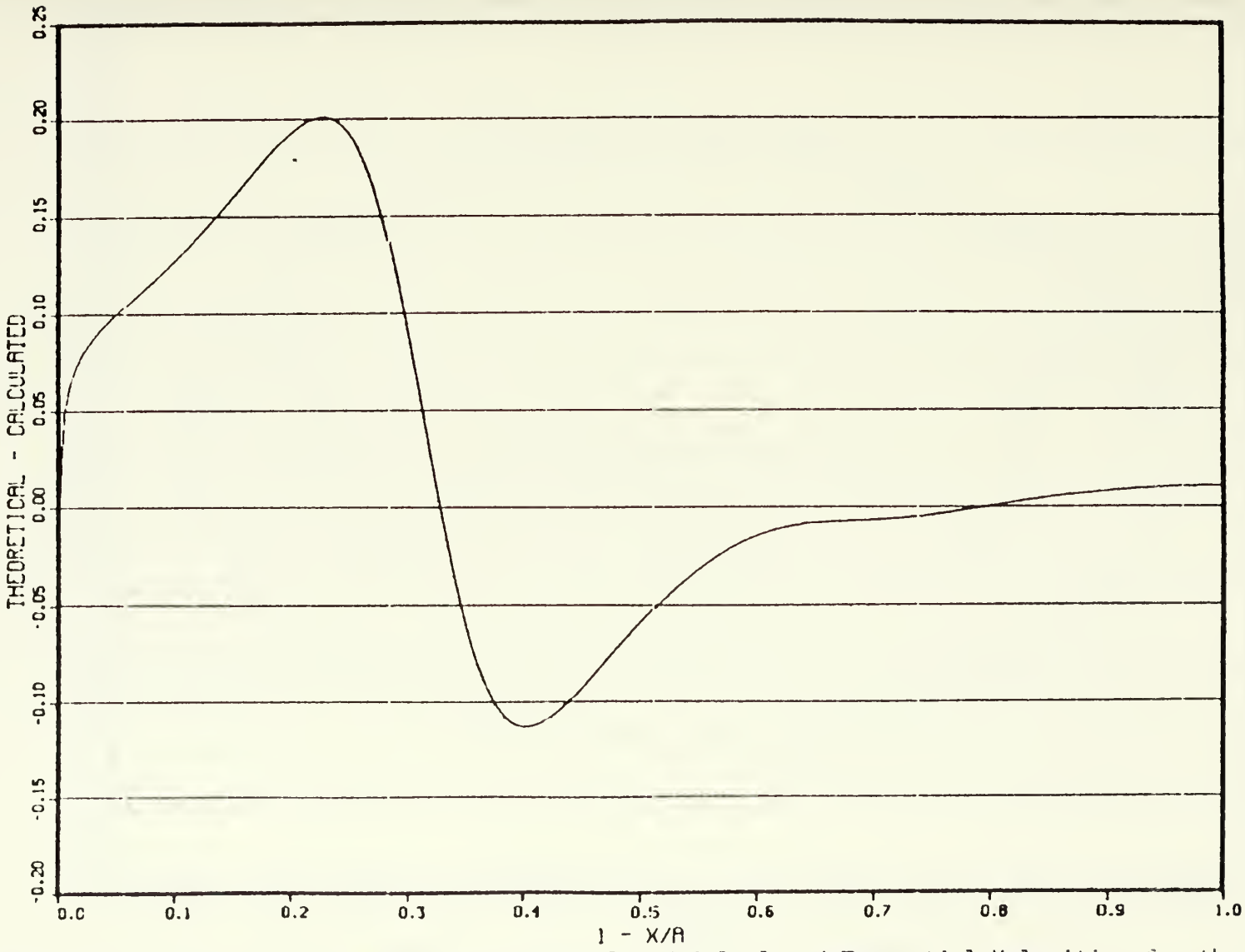


Figure 8d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 4/11

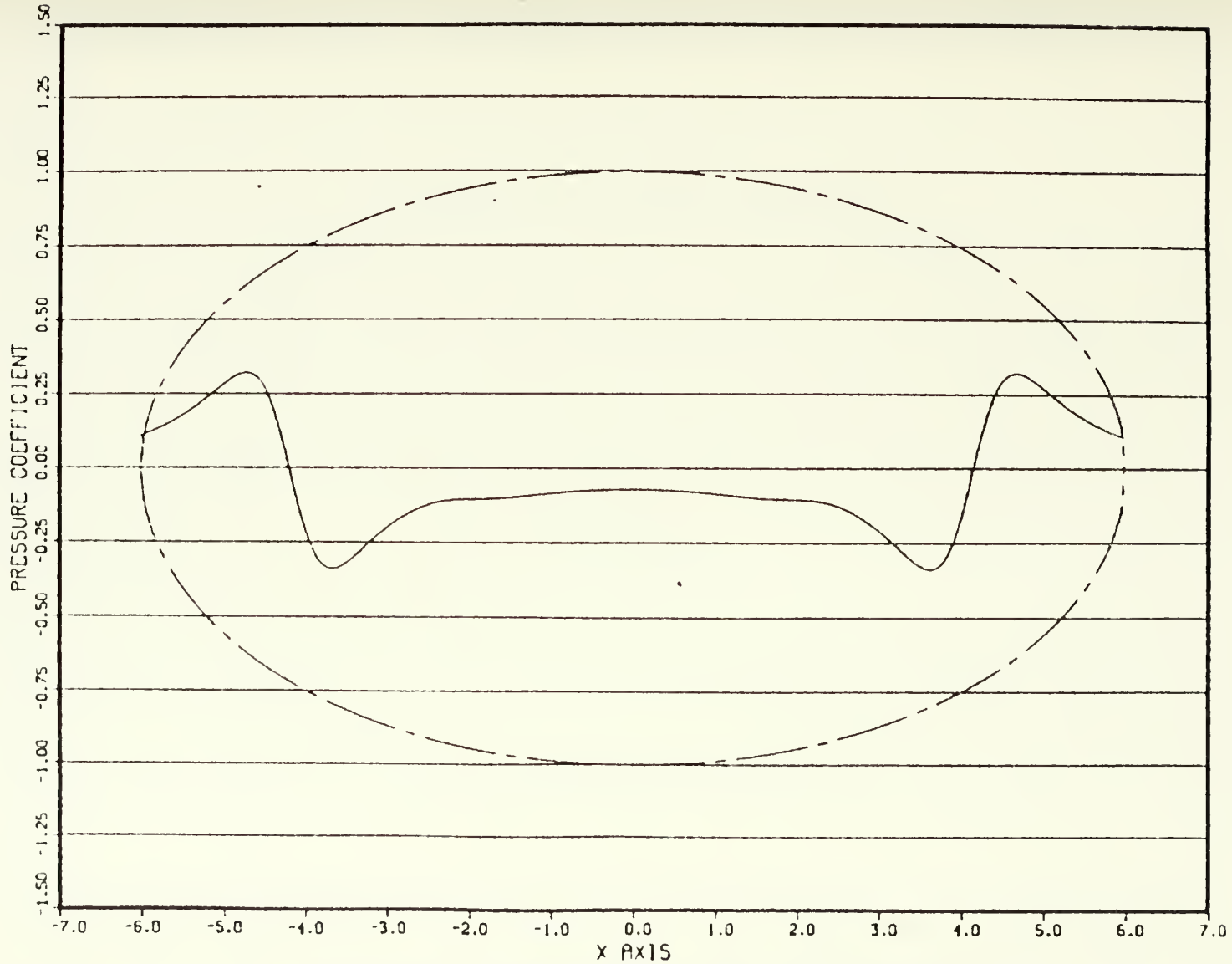


Figure 8e. Pressure Coefficient for the Case of 4/11

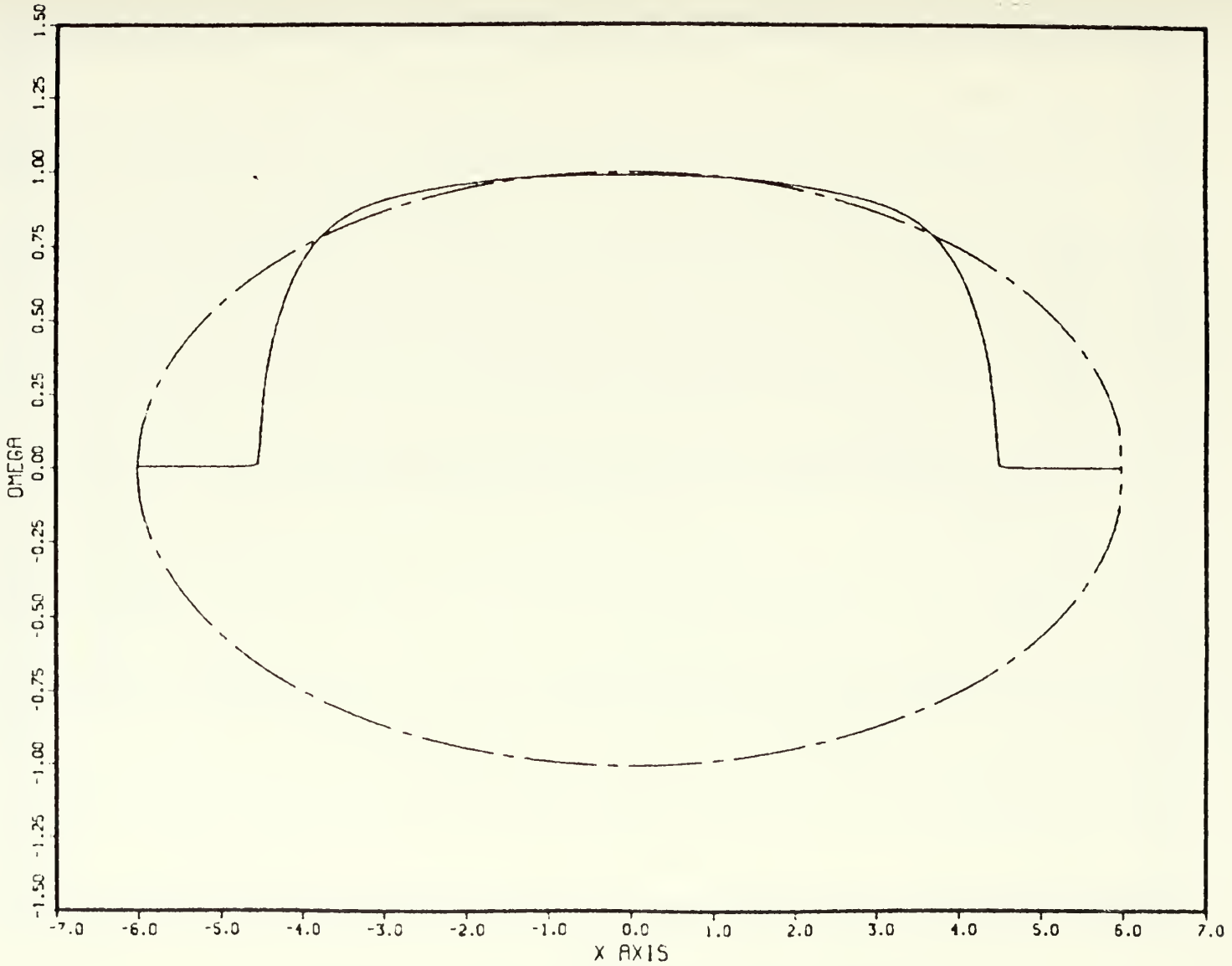


Figure 8f. Body Shape for the Case of 4/11

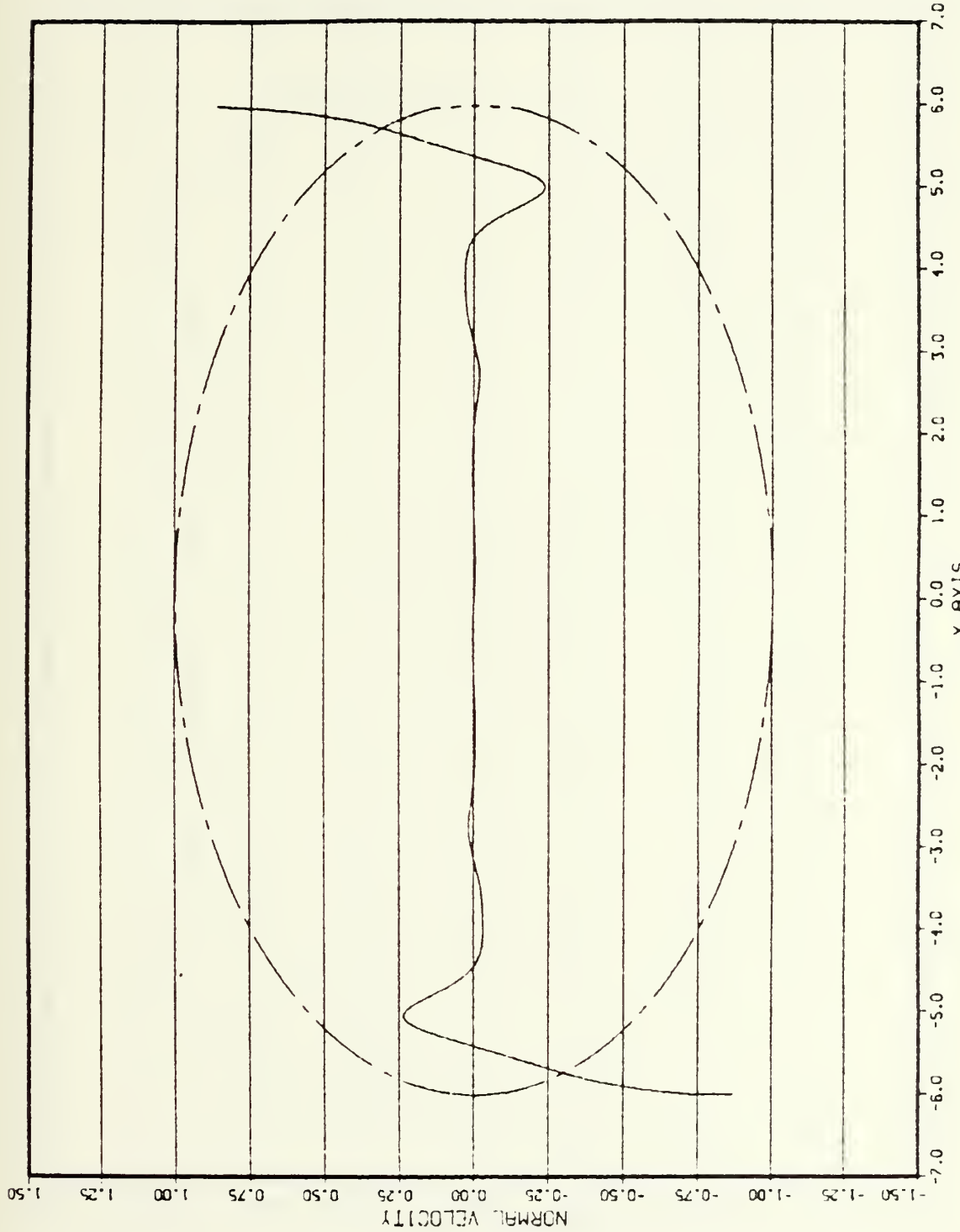


Figure 9a. Normal Velocity for the Case of 10/21

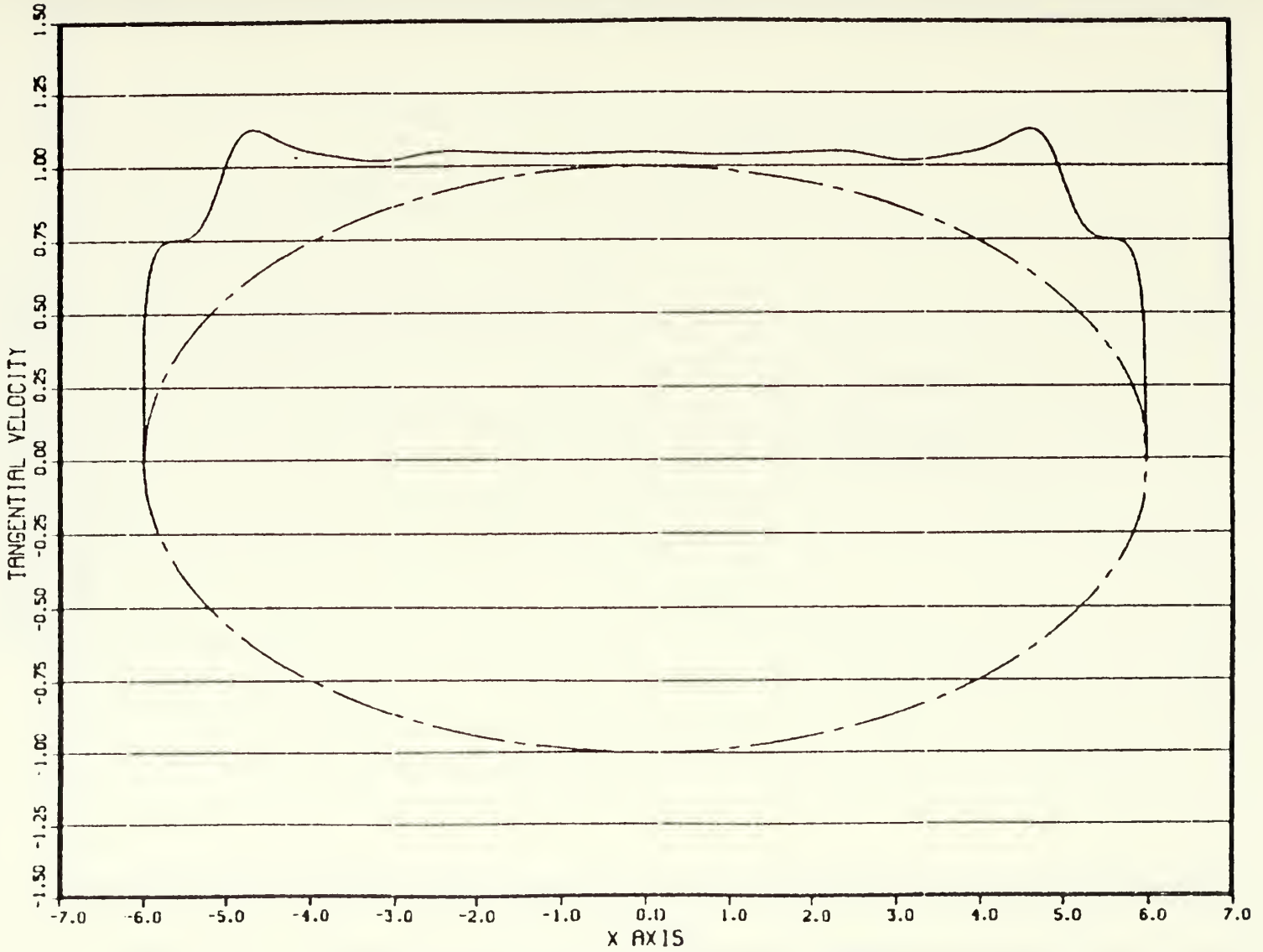


Figure 9b. Tangential Velocity for the Case of 10/21

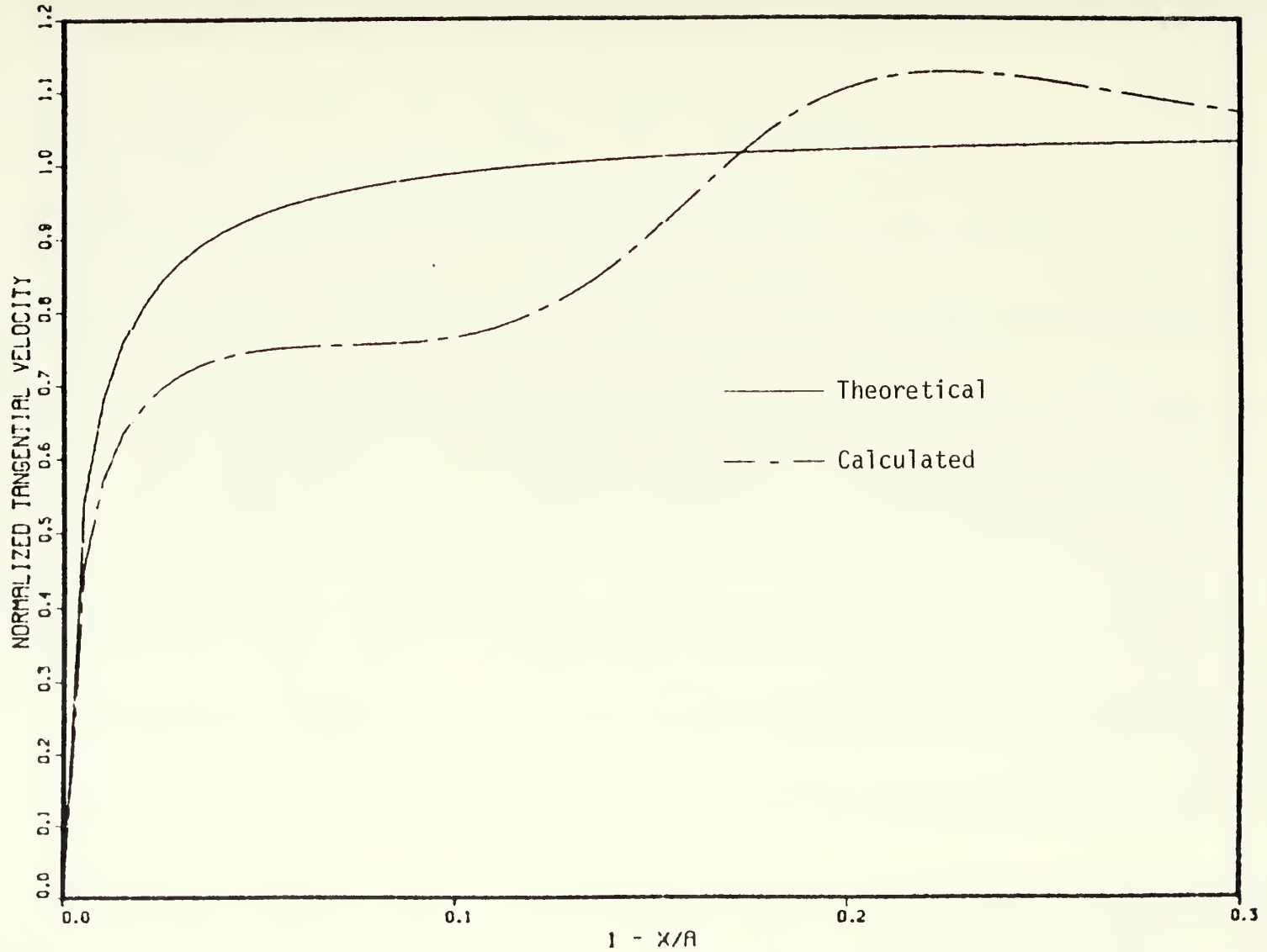


Figure 9c. Theoretical and Calculated Tangential Velocities for the Case of 10/21

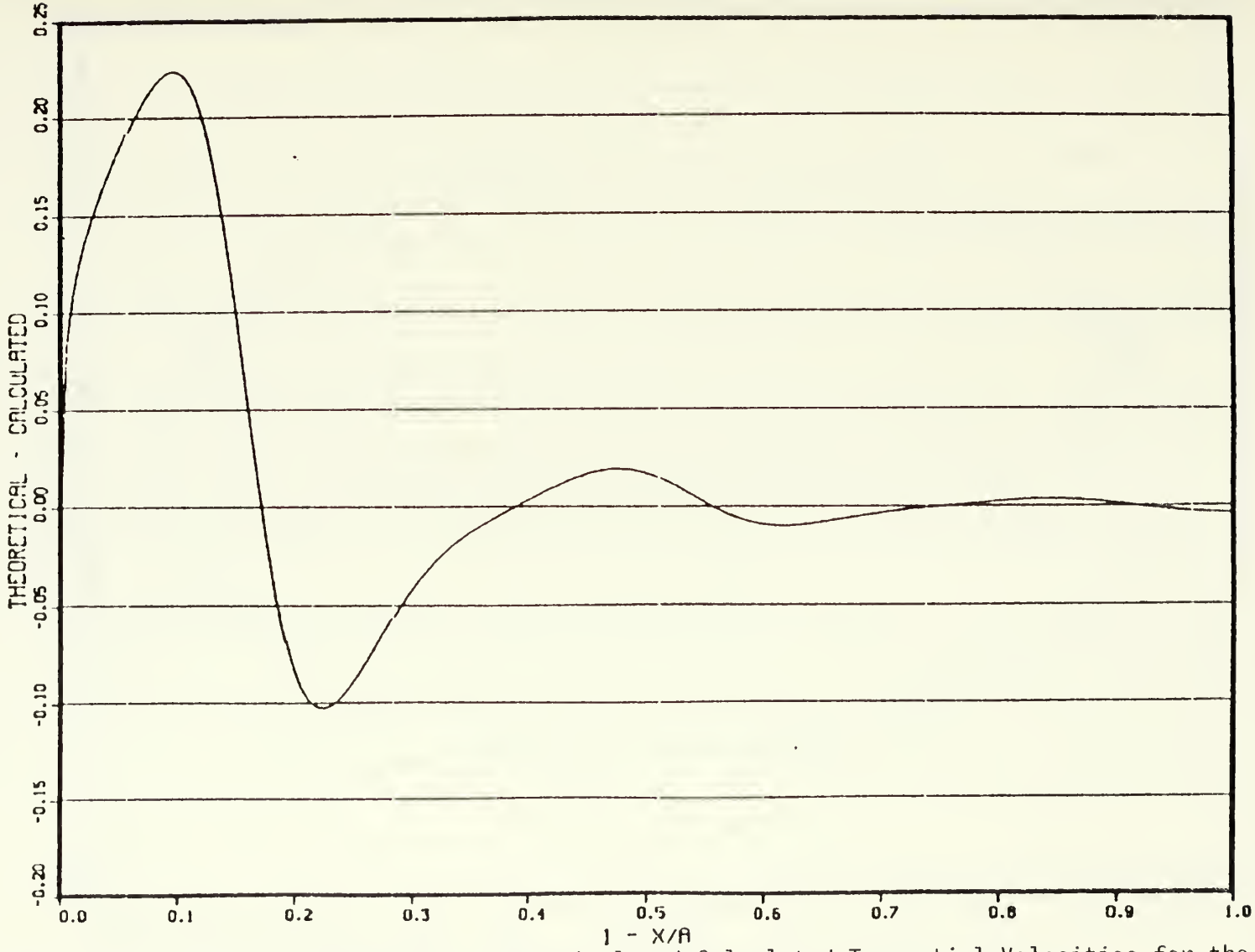


Figure 9d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 10/21

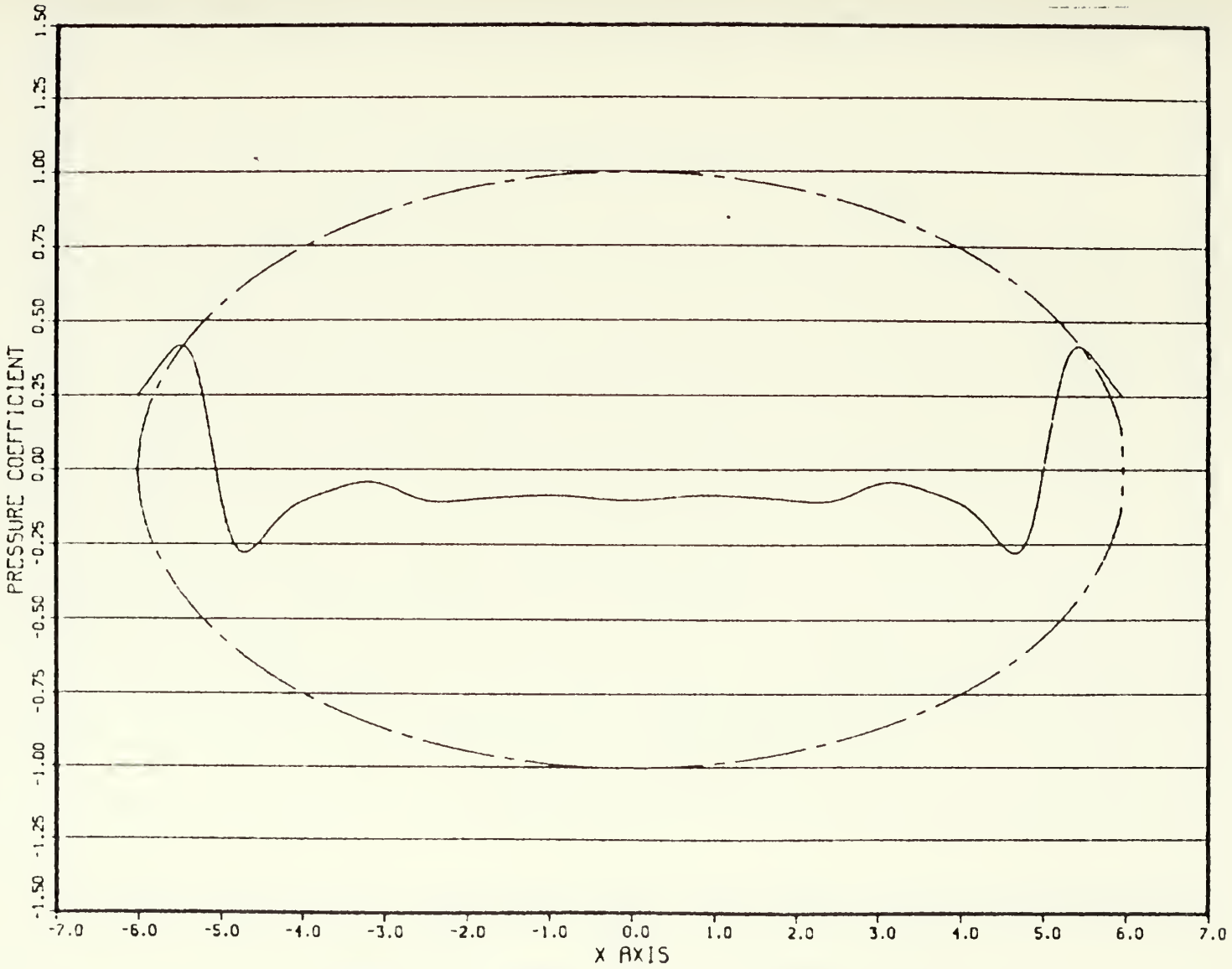


Figure 9e. Pressure Coefficient for the Case of 10/21

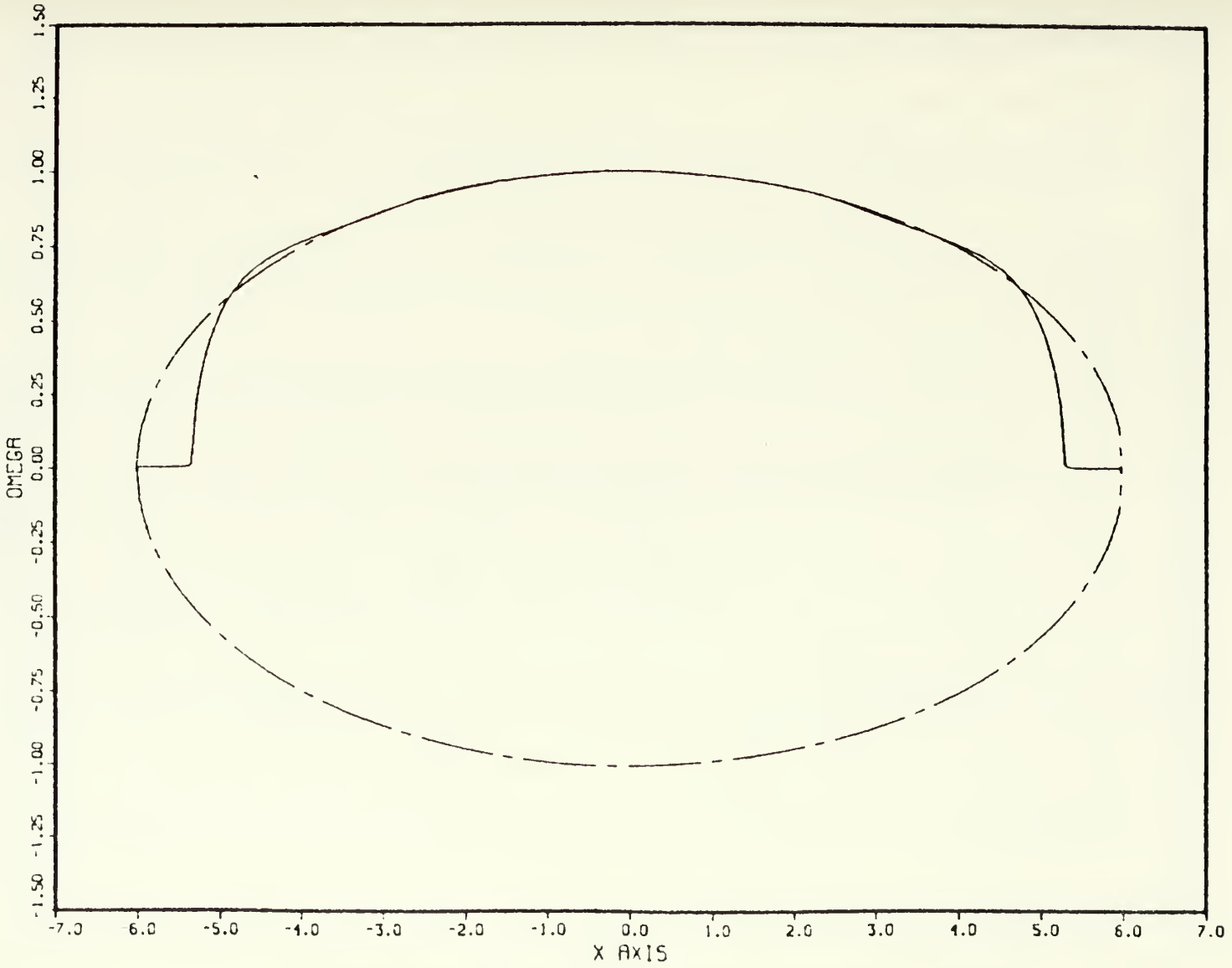


Figure 9f. Body Shape for the Case of 10/21

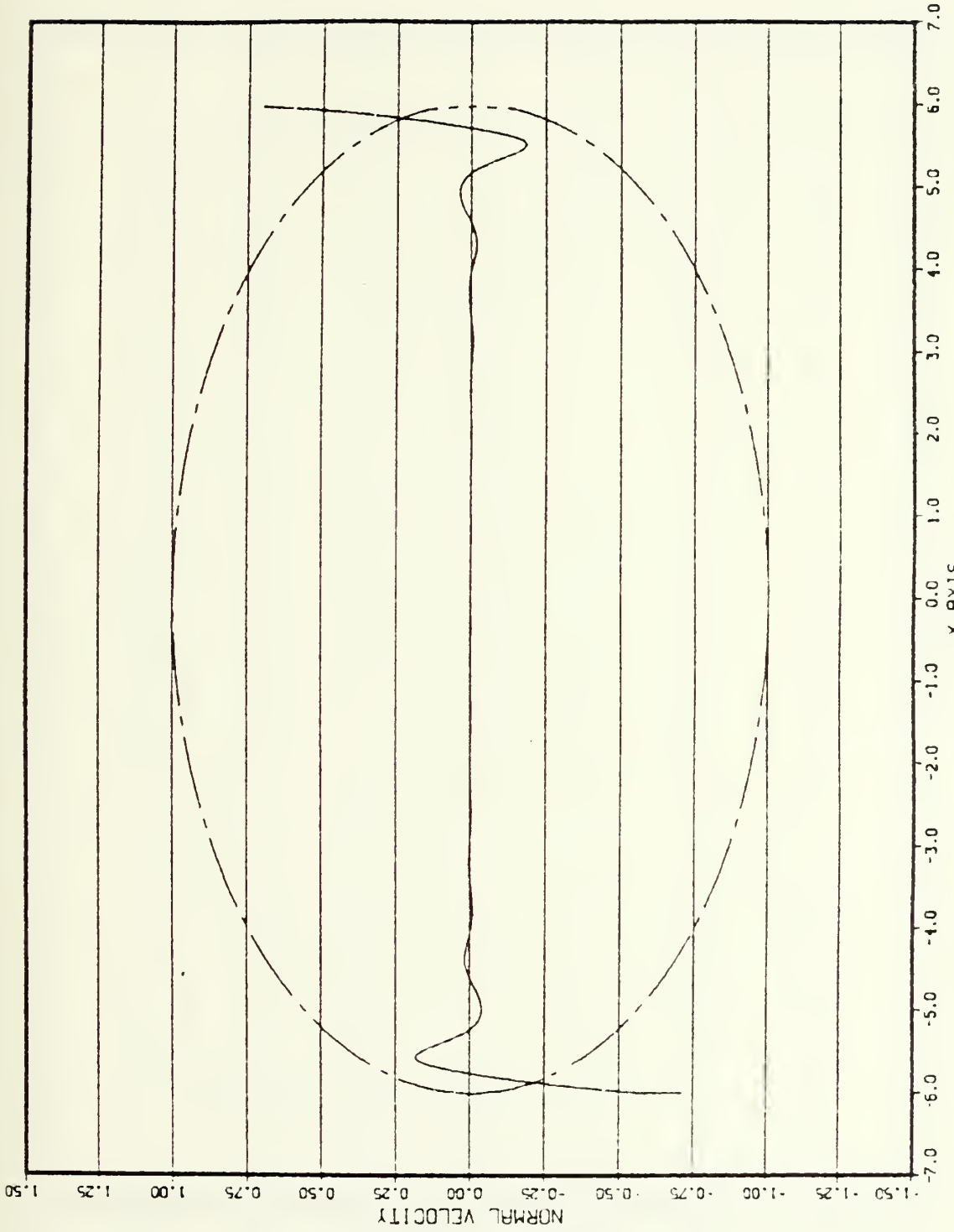


Figure 10a. Normal Velocity for the Case of 20/41

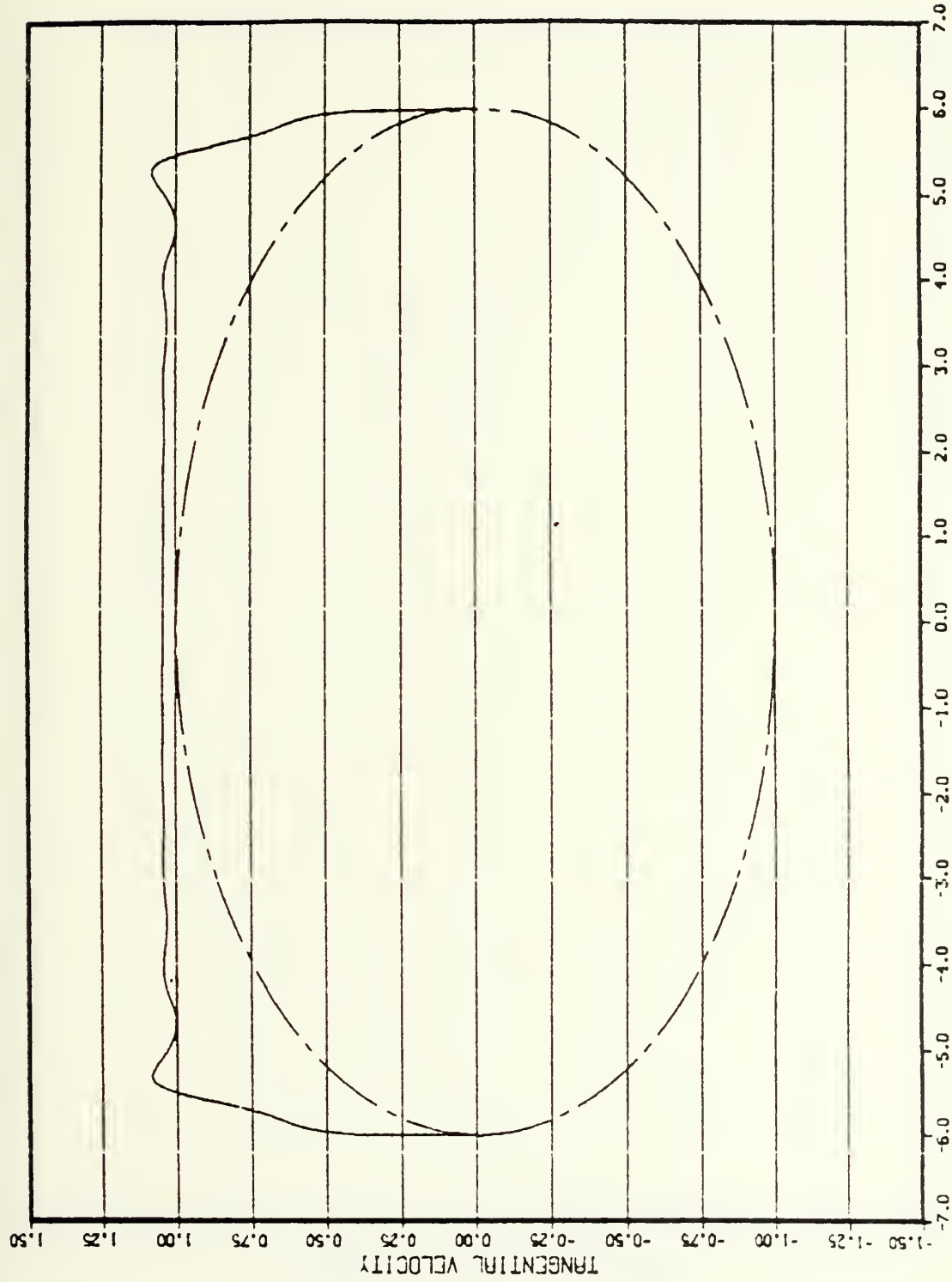


Figure 10b. Tangential Velocity for the Case of 20/41

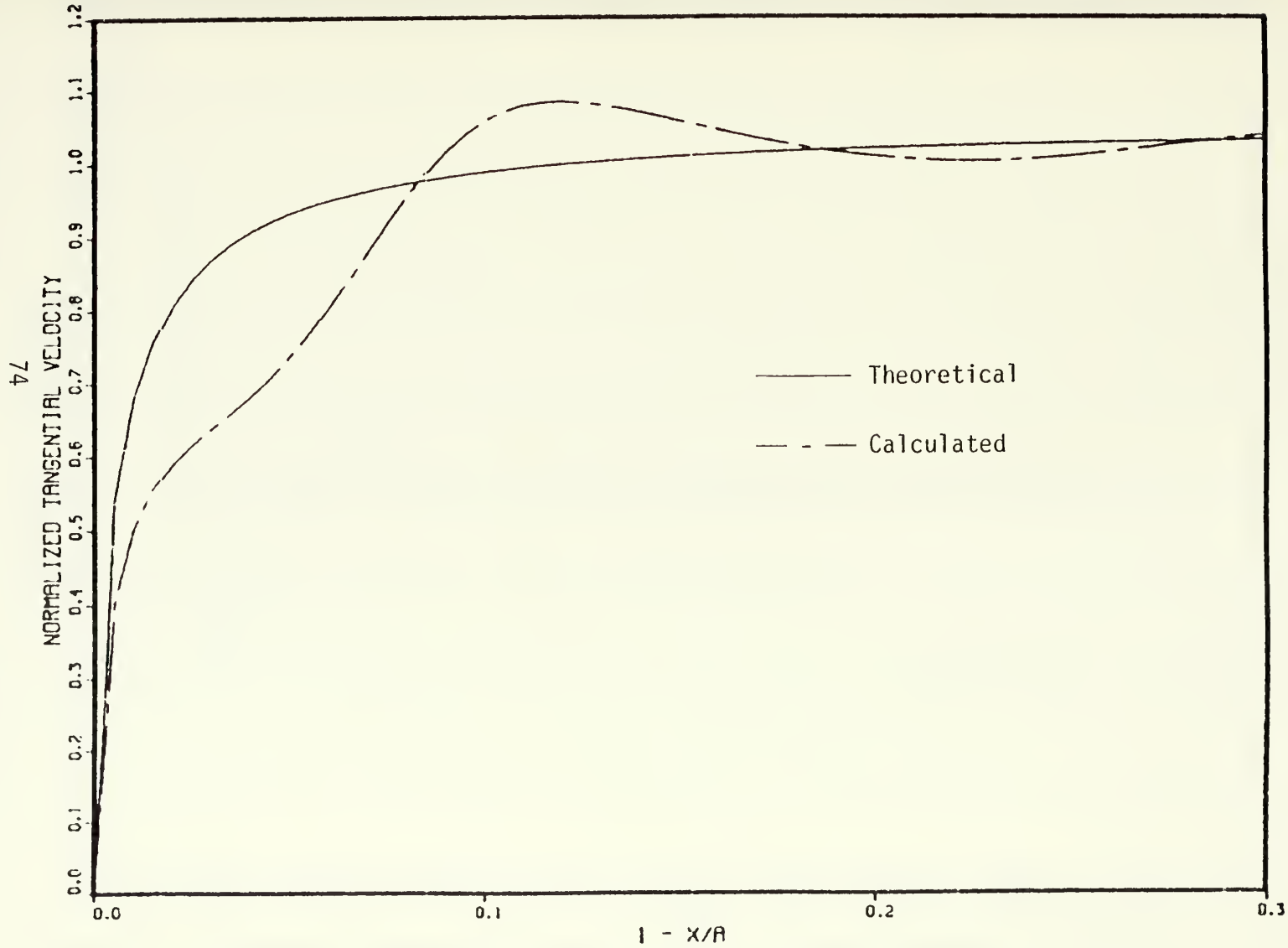


Figure 10c. Theoretical and Calculated Tangential Velocities for the Case of 20/41

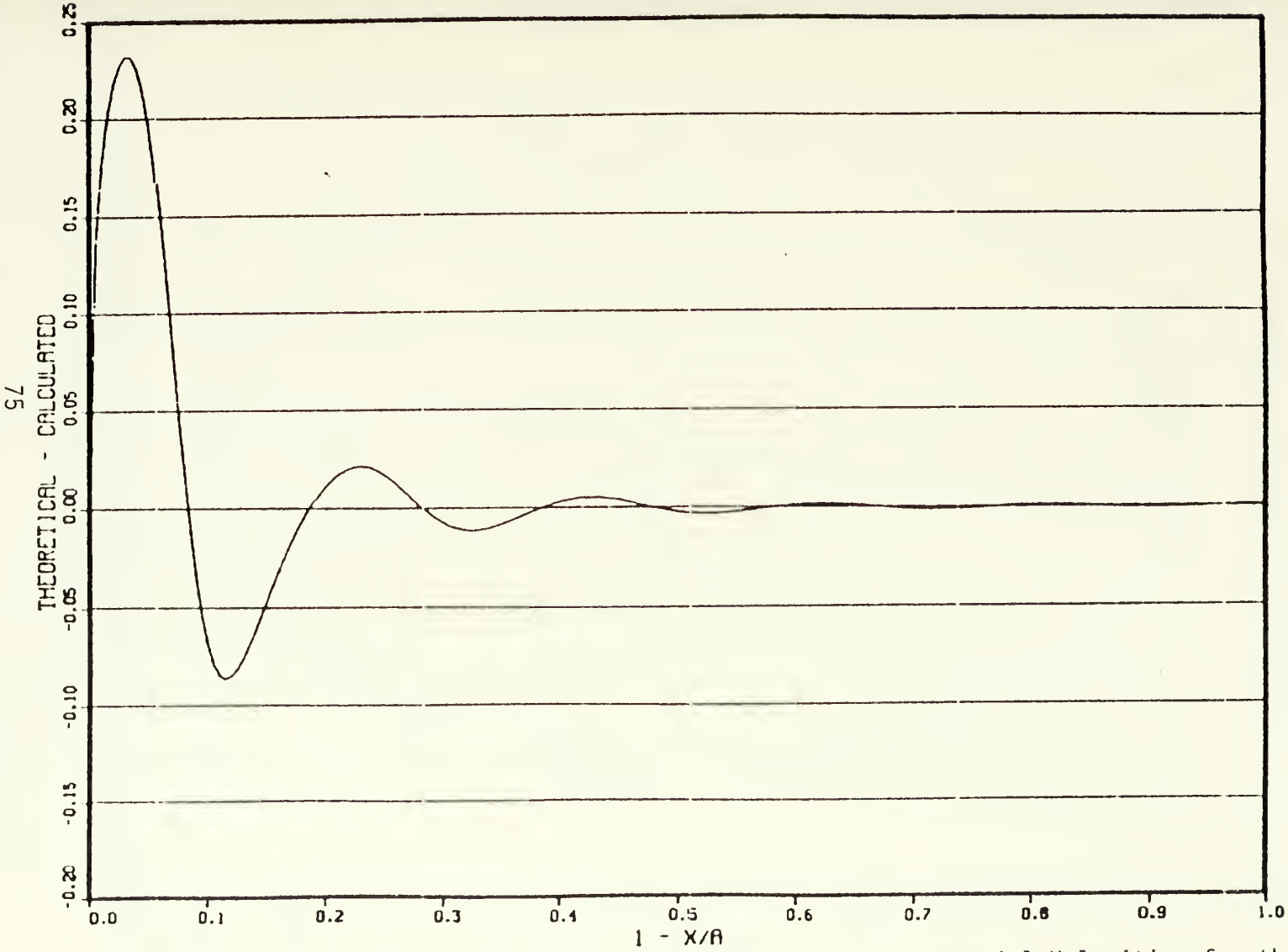


Figure 10d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 20/41

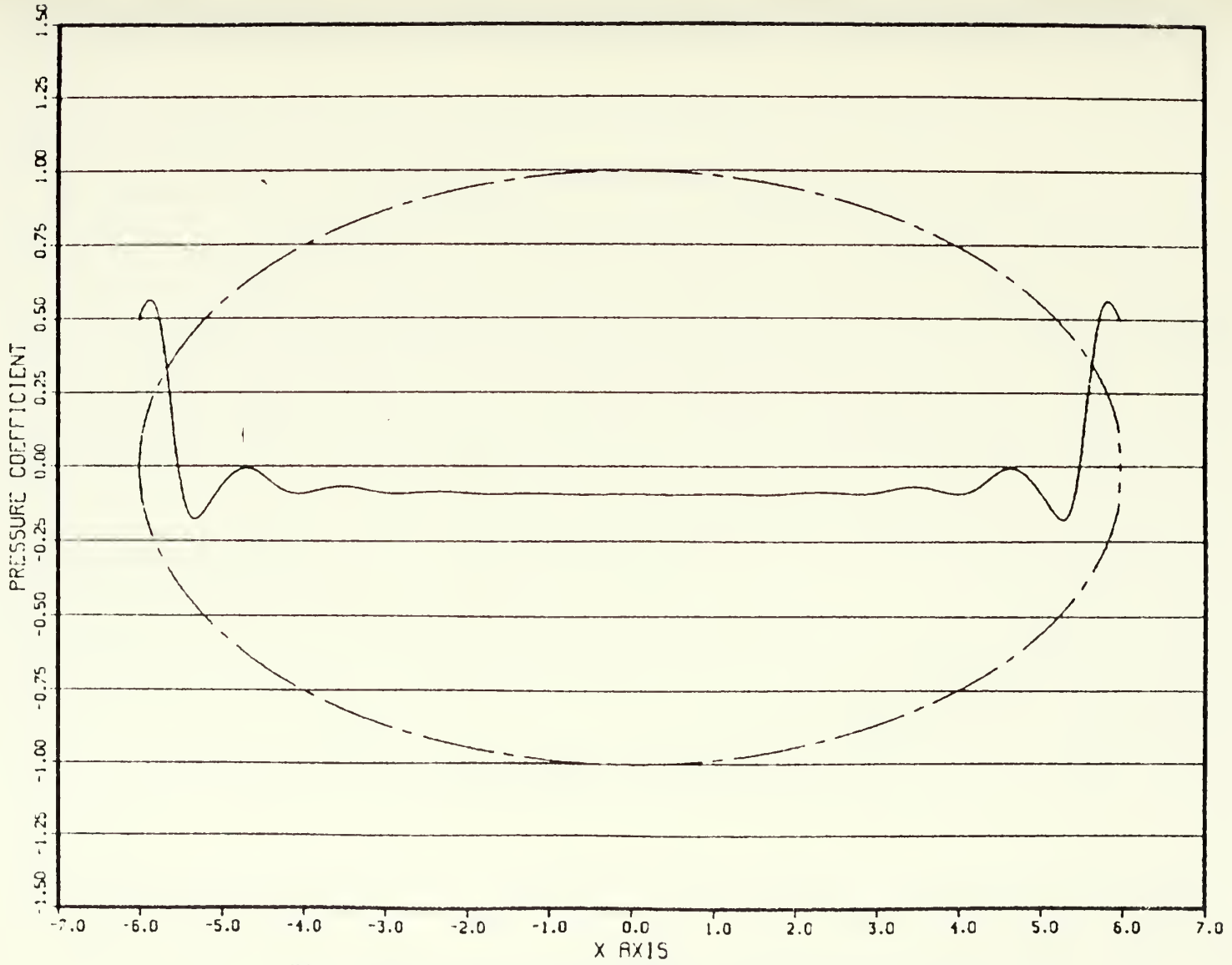


Figure 10e. Pressure Coefficient for the Case of 20/41

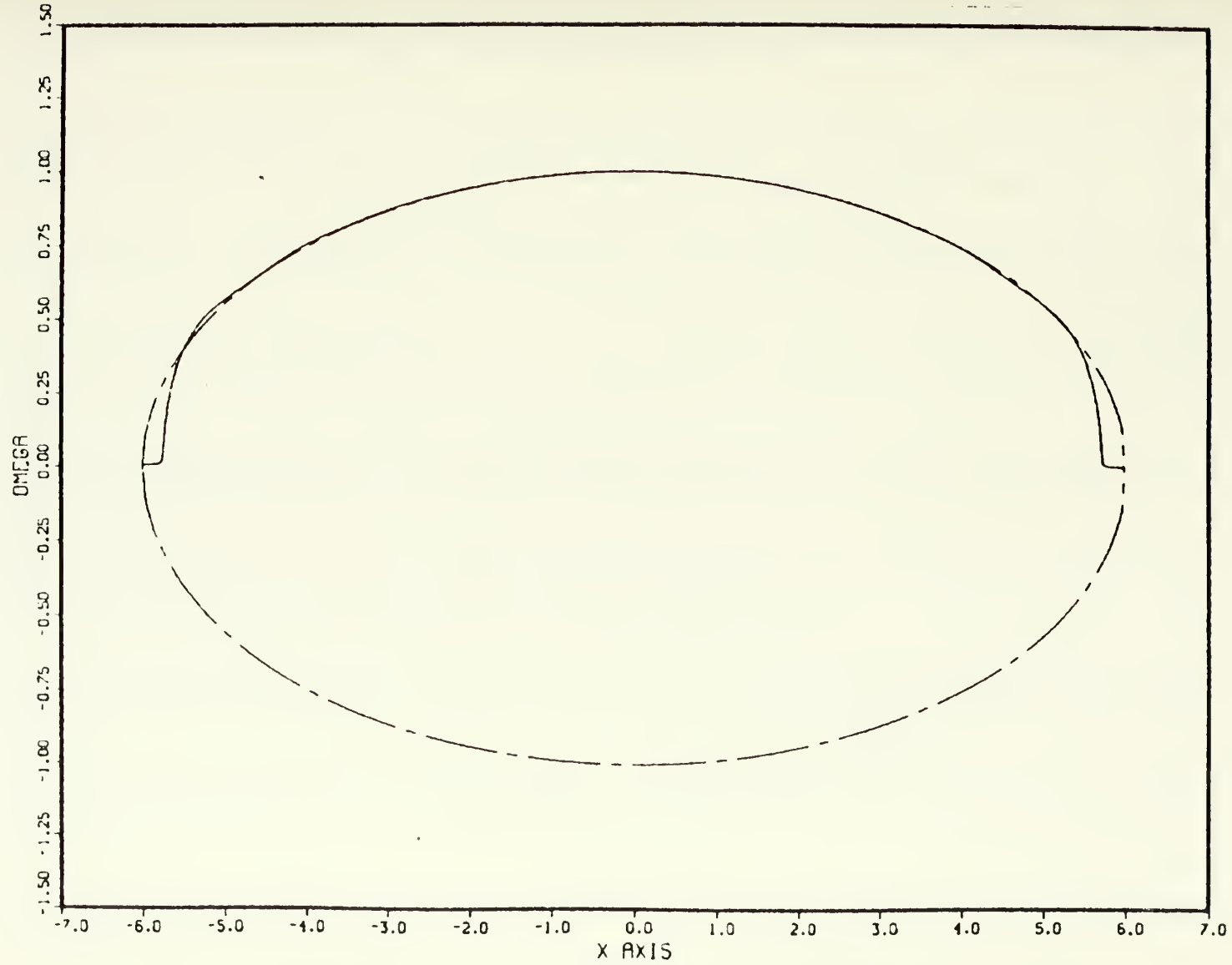


Figure 10f. Body Shape for the Case of 20/41

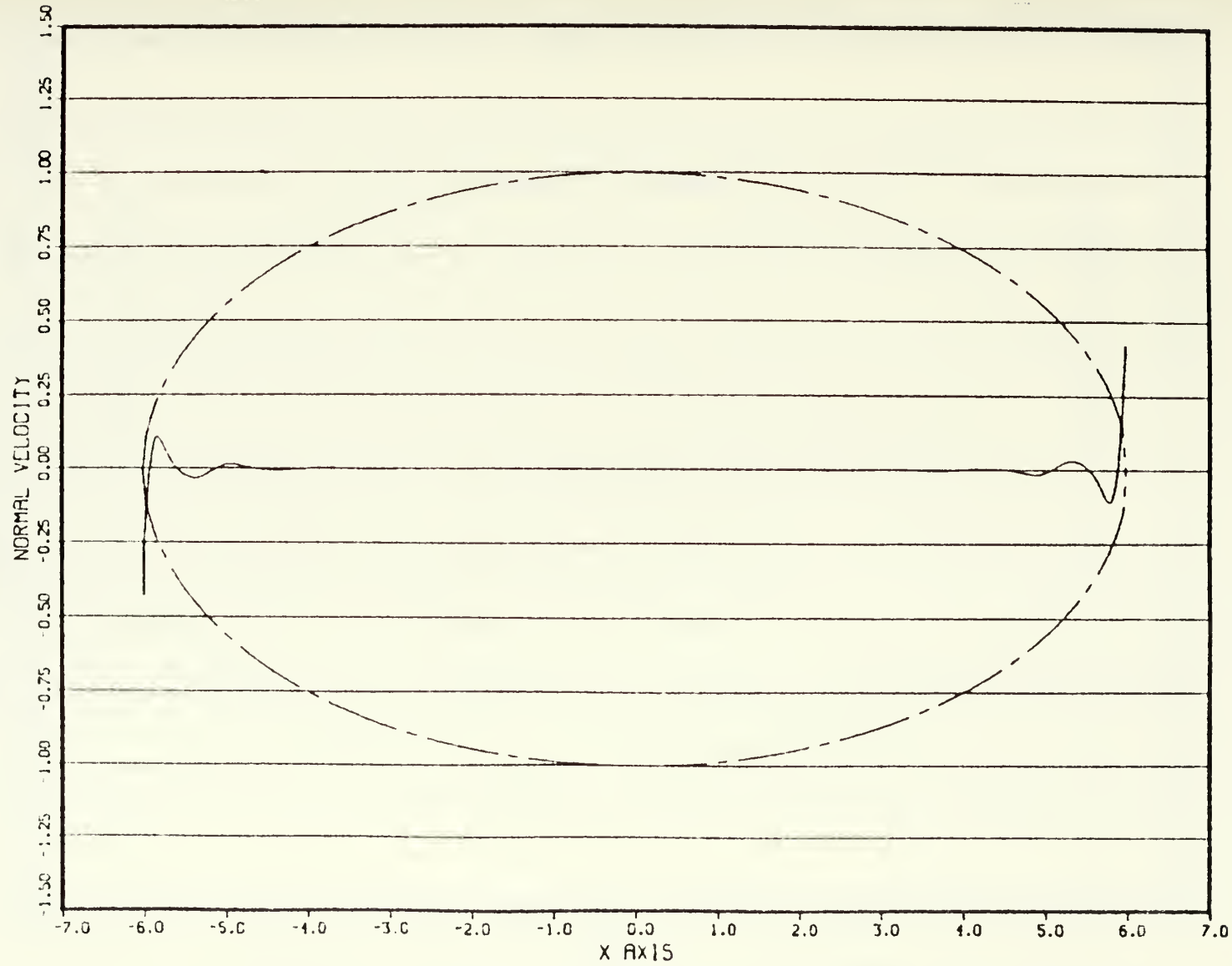


Figure 11a. Normal Velocity for the Modified Case of 20/41

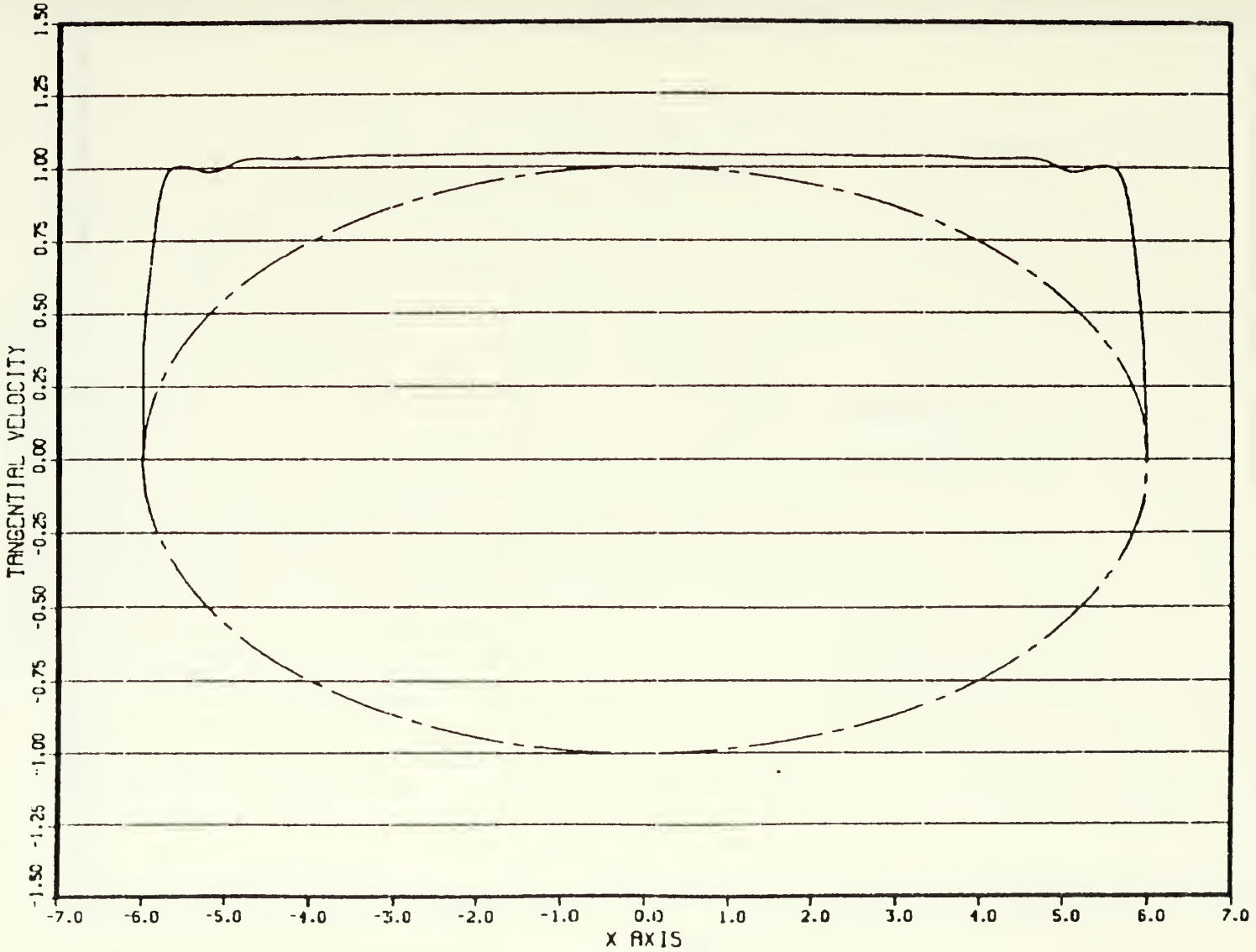


Figure 11b. Tangential Velocity for the Modified Case of 20/41

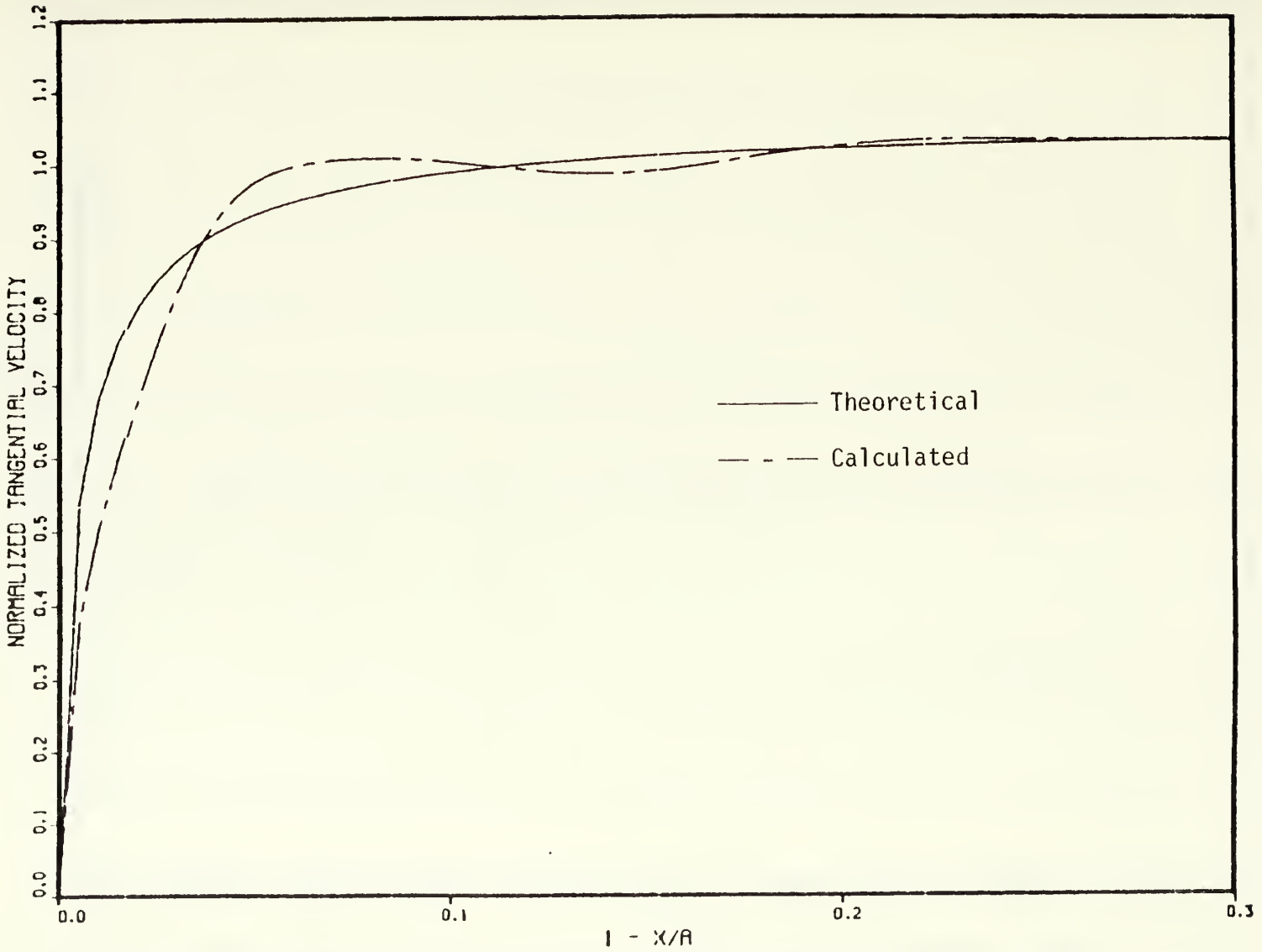


Figure 11c. Theoretical and Calculated Tangential Velocities for the Modified Case of 20/41

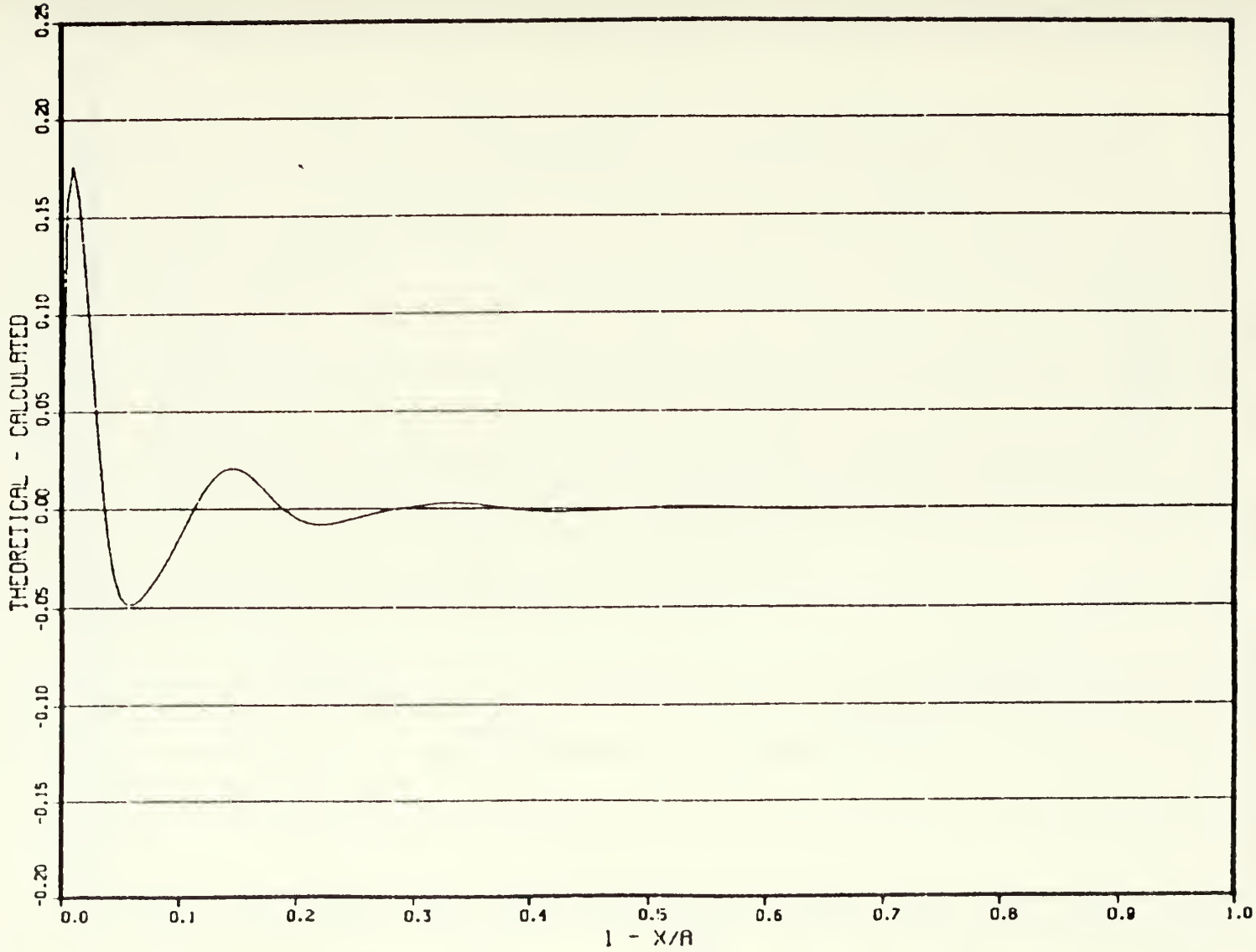


Figure 11d. Difference between the Theoretical and Calculated Tangential Velocities for the Modified Case of 20/

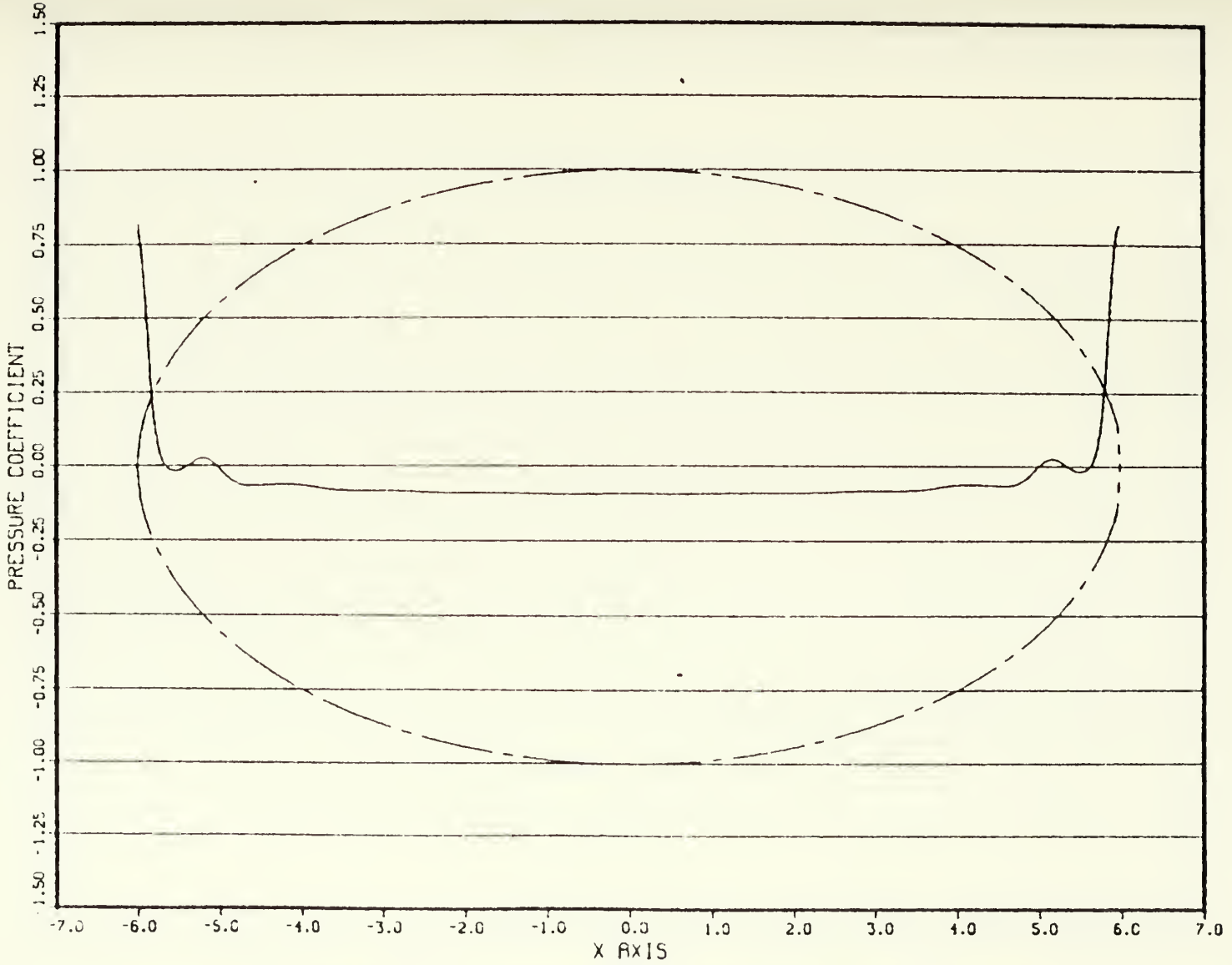


Figure 11e. Pressure Coefficient for the Modified Case of 20/41

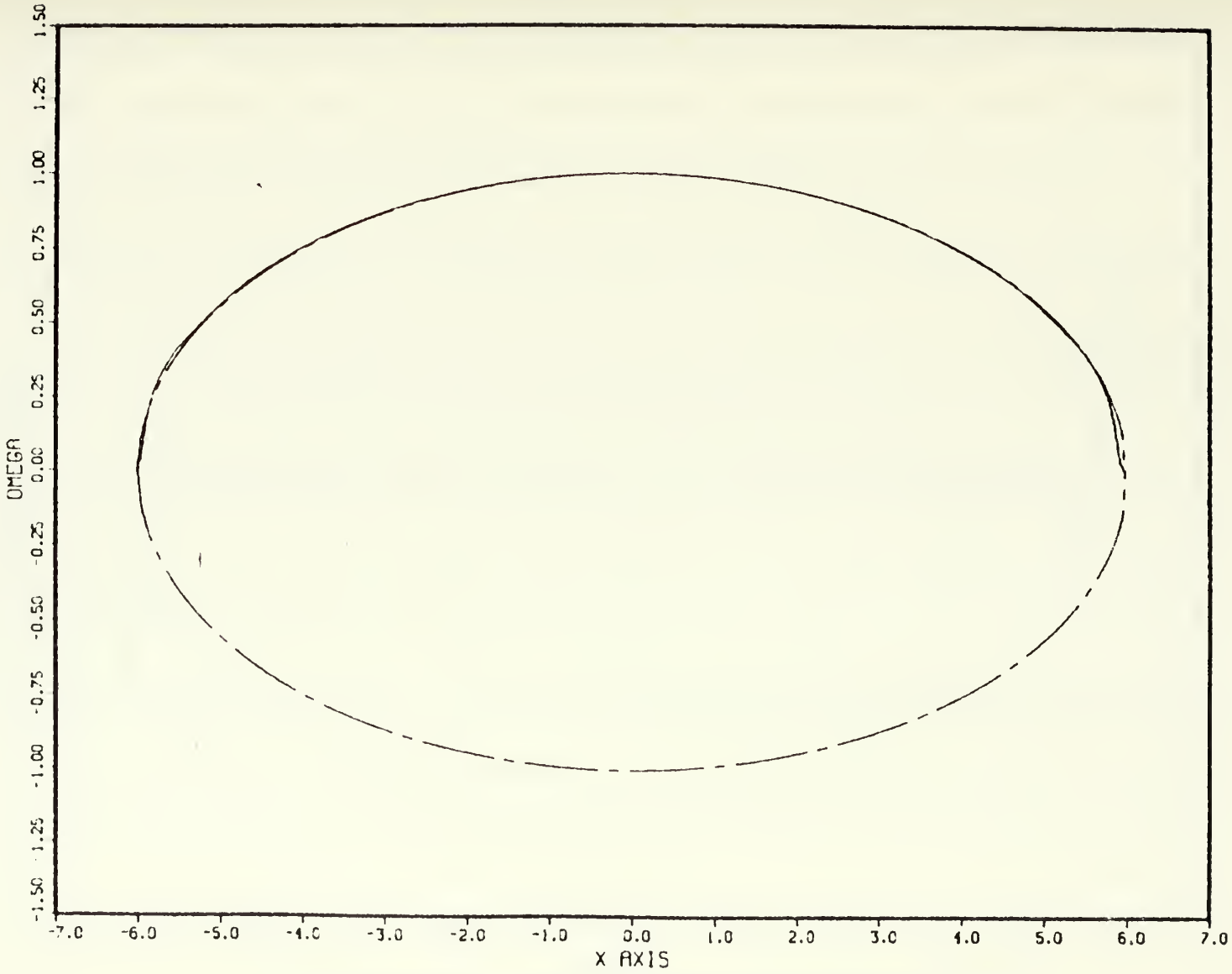


Figure 11f. Body Shape for the Modified Case of 20/41

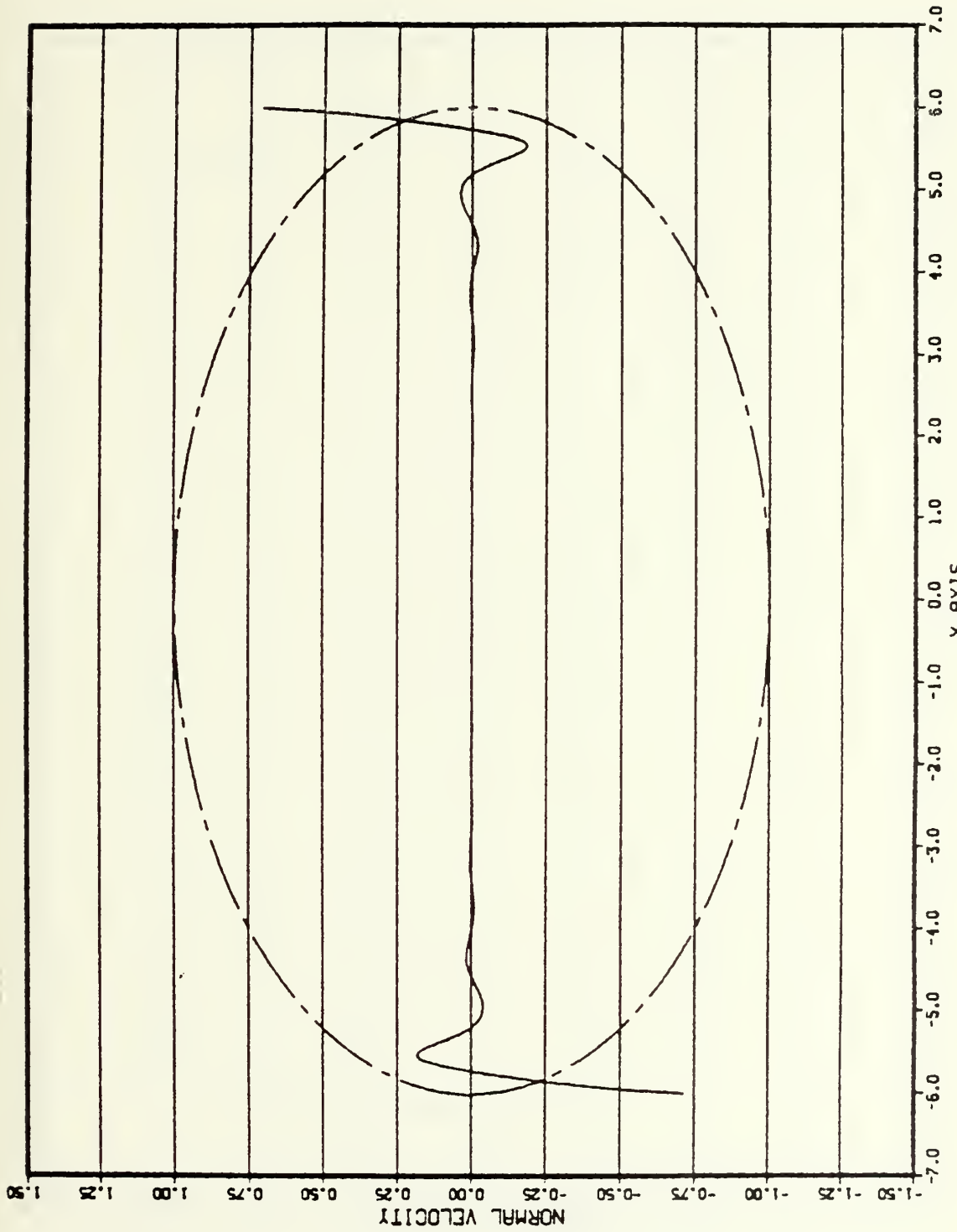


Figure 12a. Normal Velocity for the Case of 20/40

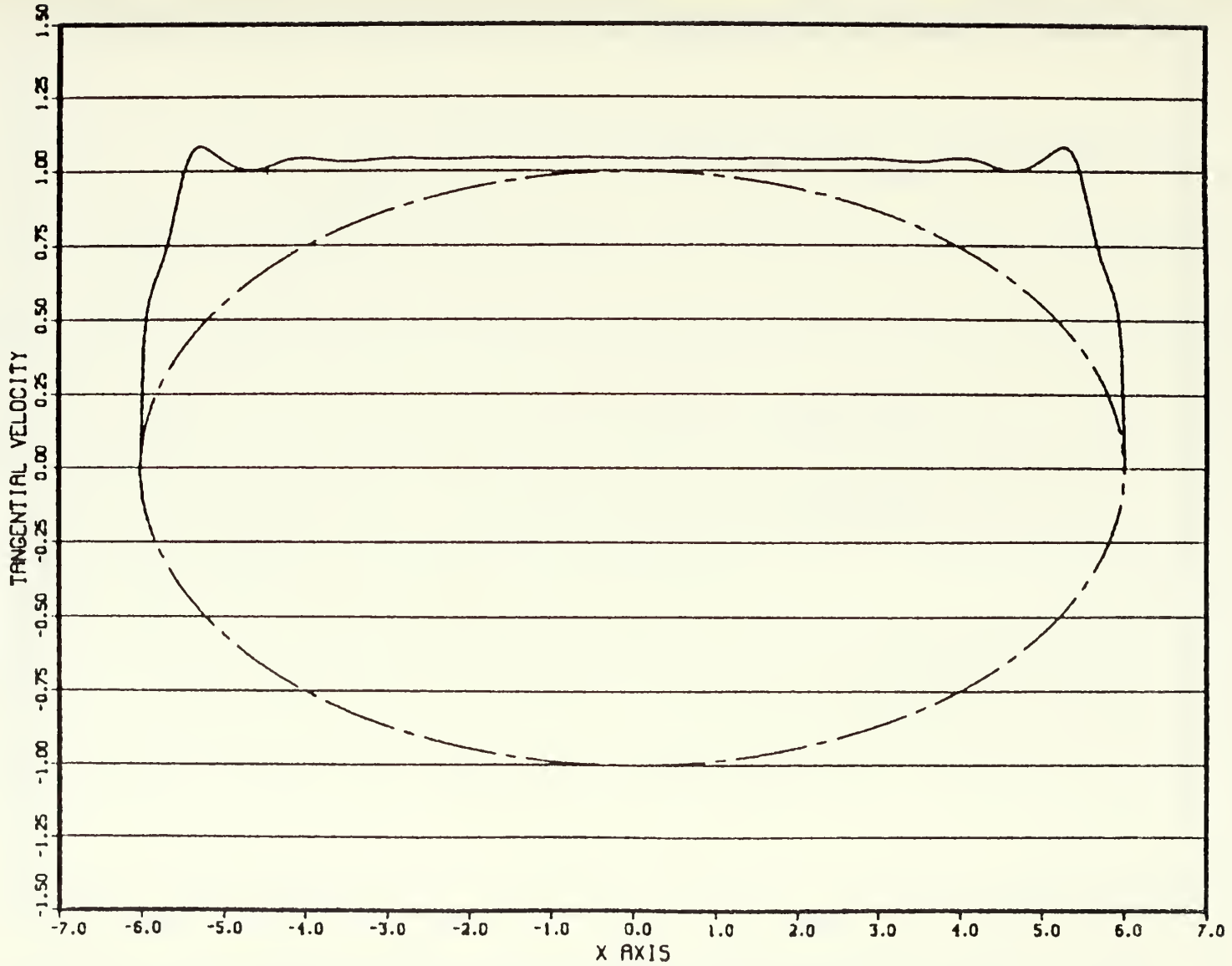


Figure 12b. Tangential Velocity for the Case of 20/40

Figure 12c. Theoretical and Calculated Tangential Velocities for the Case of 20/40

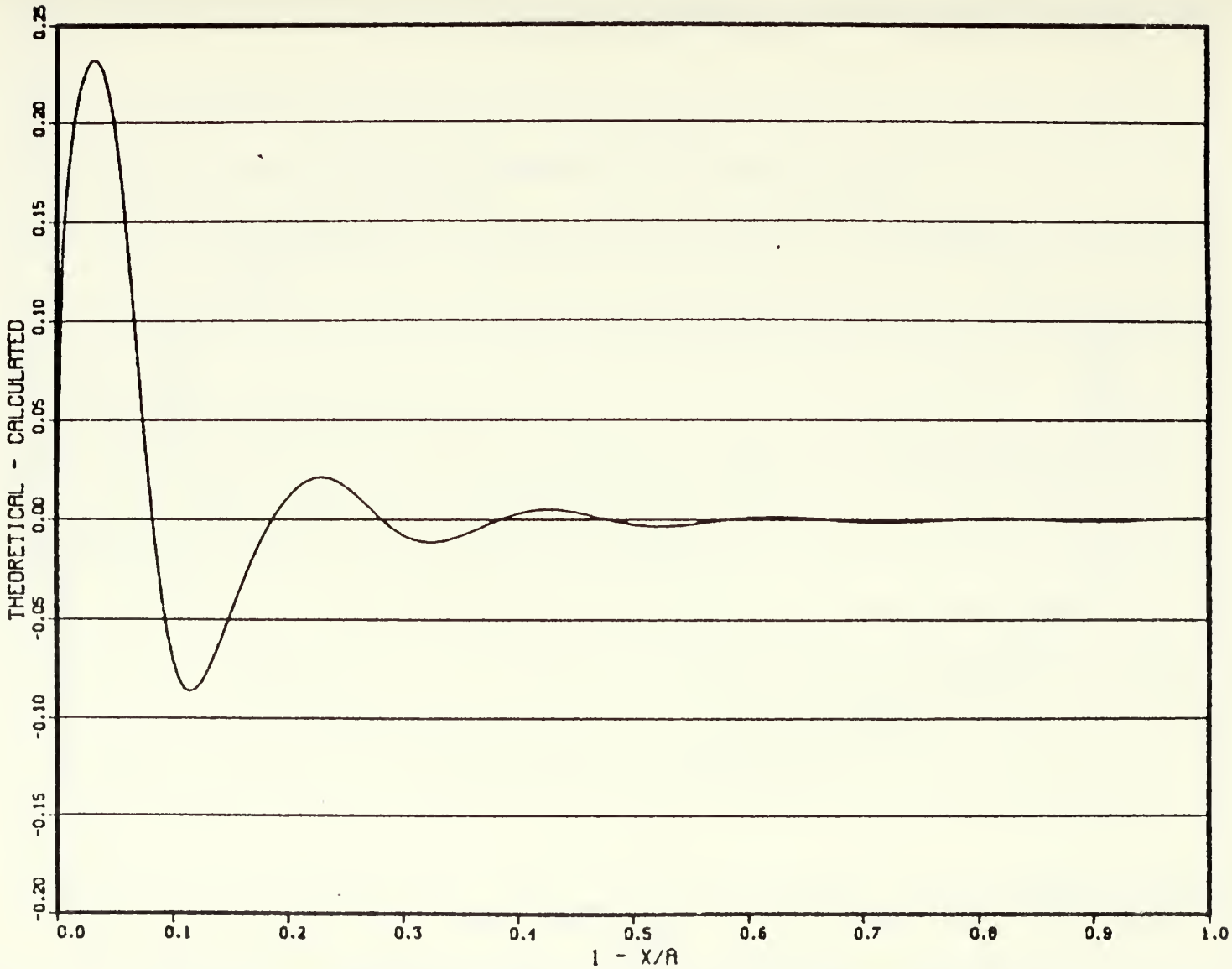


Figure 12d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 20/40

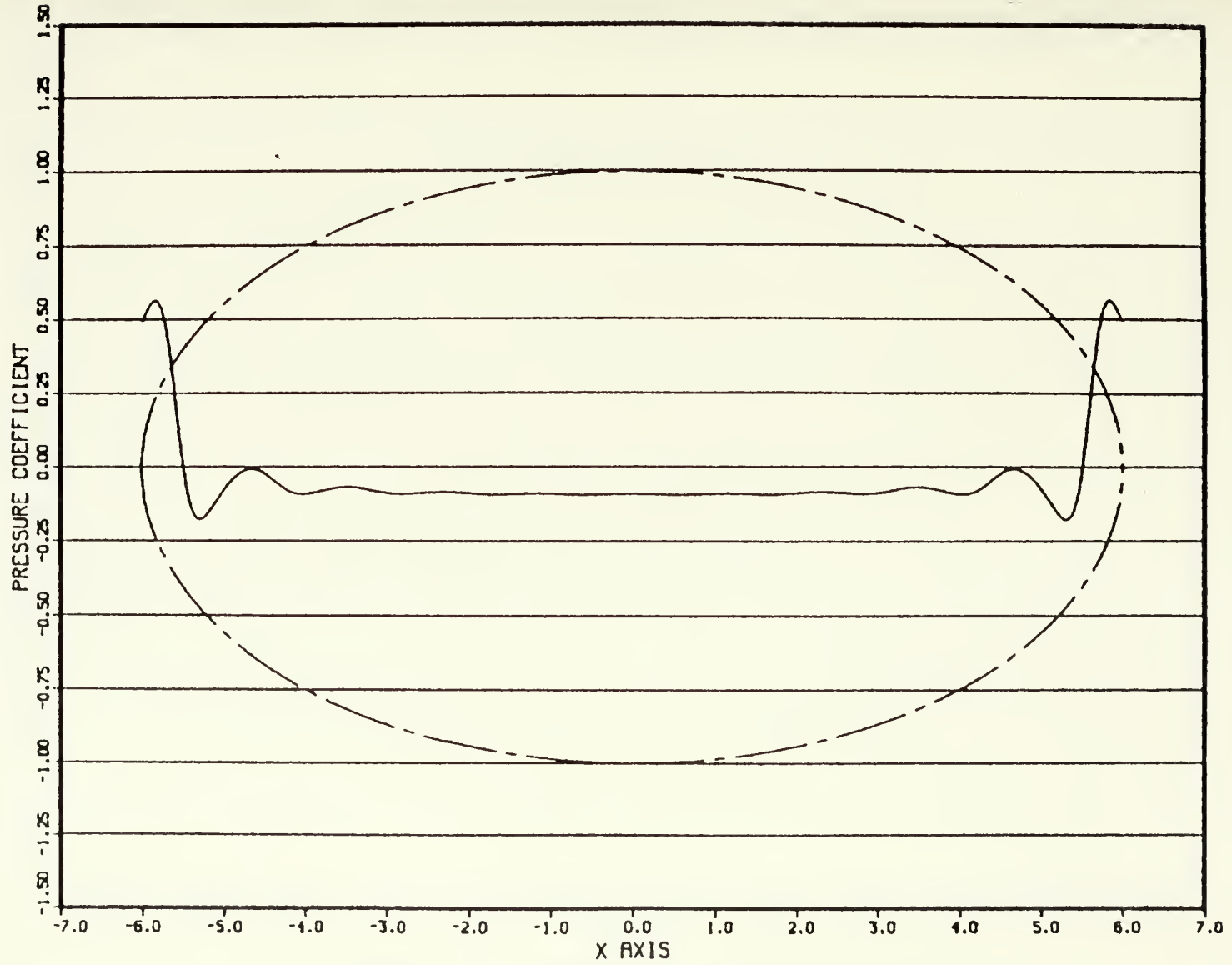


Figure 12e. Pressure Coefficient for the Case of 20/40

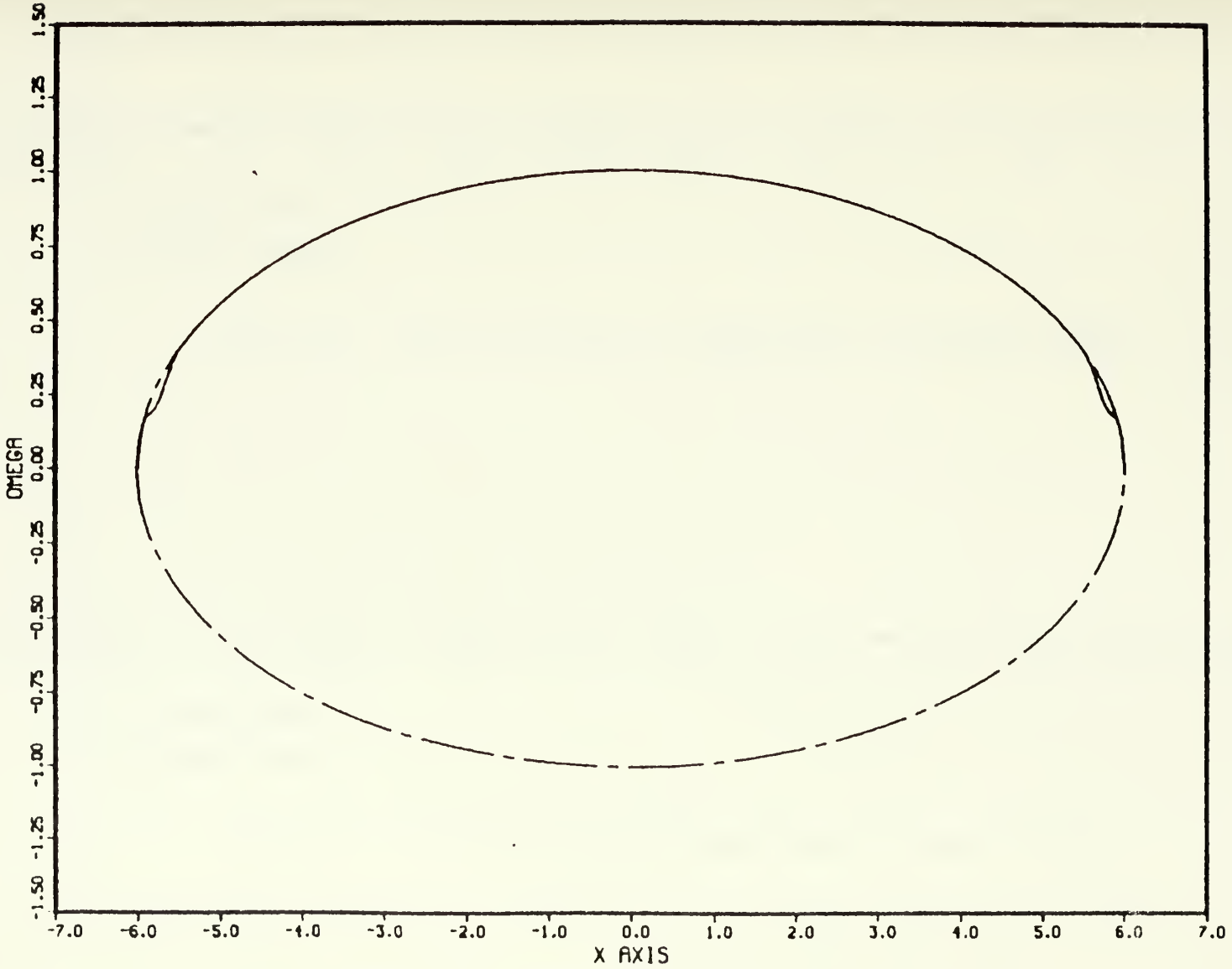
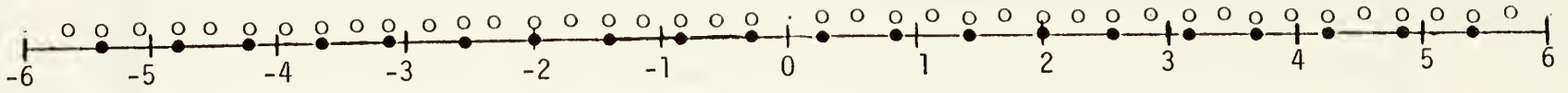


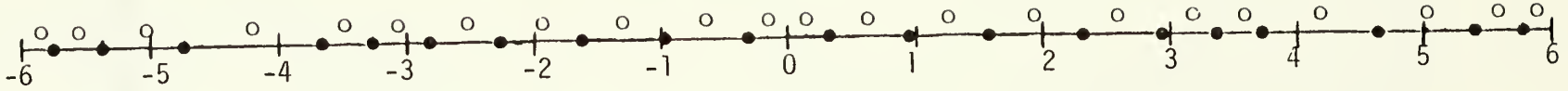
Figure 12f. Body Shape for the Case of 20/40



• Singularities

o Control Points

Figure 13a. Initial Positions of Singularities and Control Points



• Singularities

o Control Points

Figure 13b. Final Positions of Singularities and Control Points

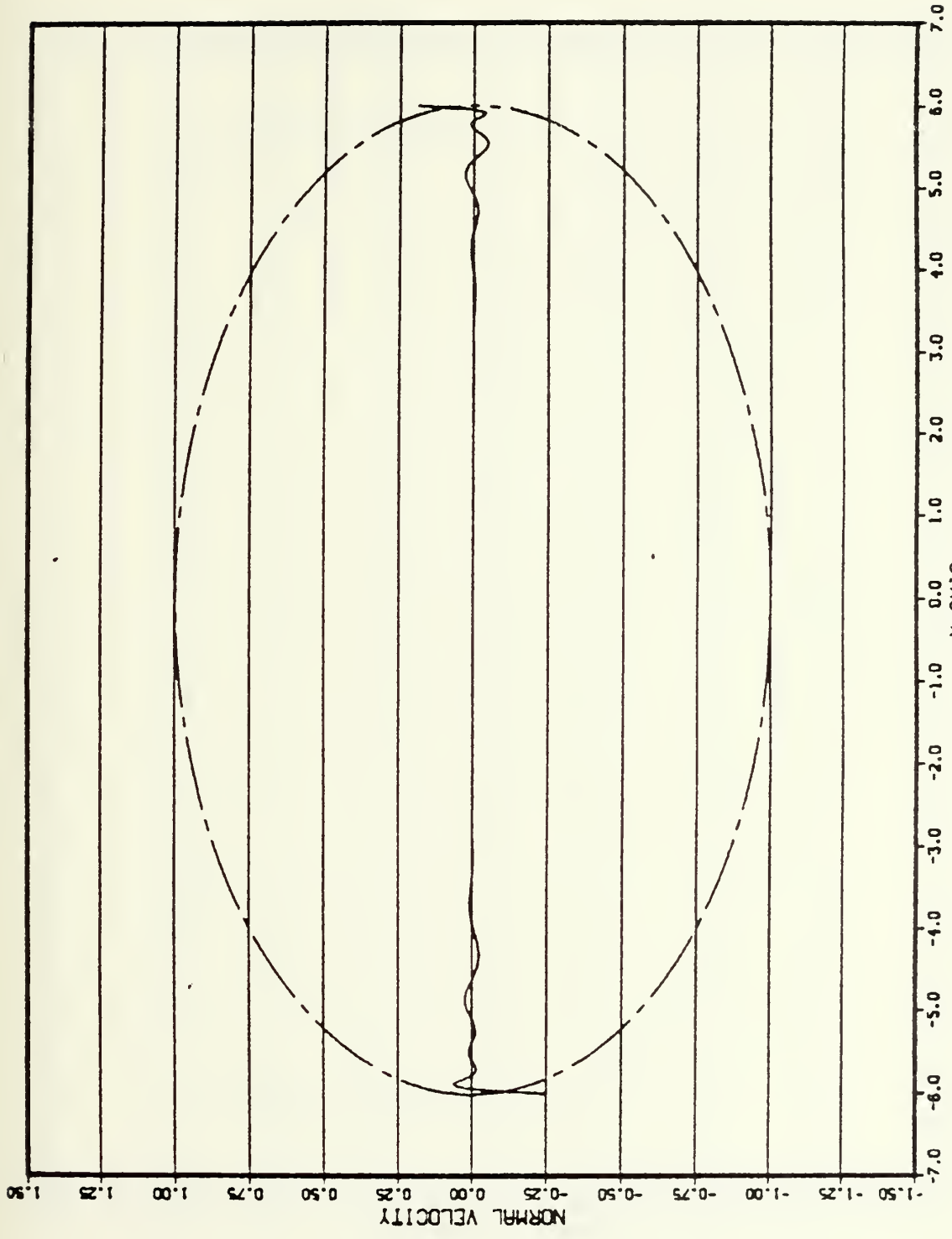


Figure 14a. Normal Velocity for the Case of 20/22

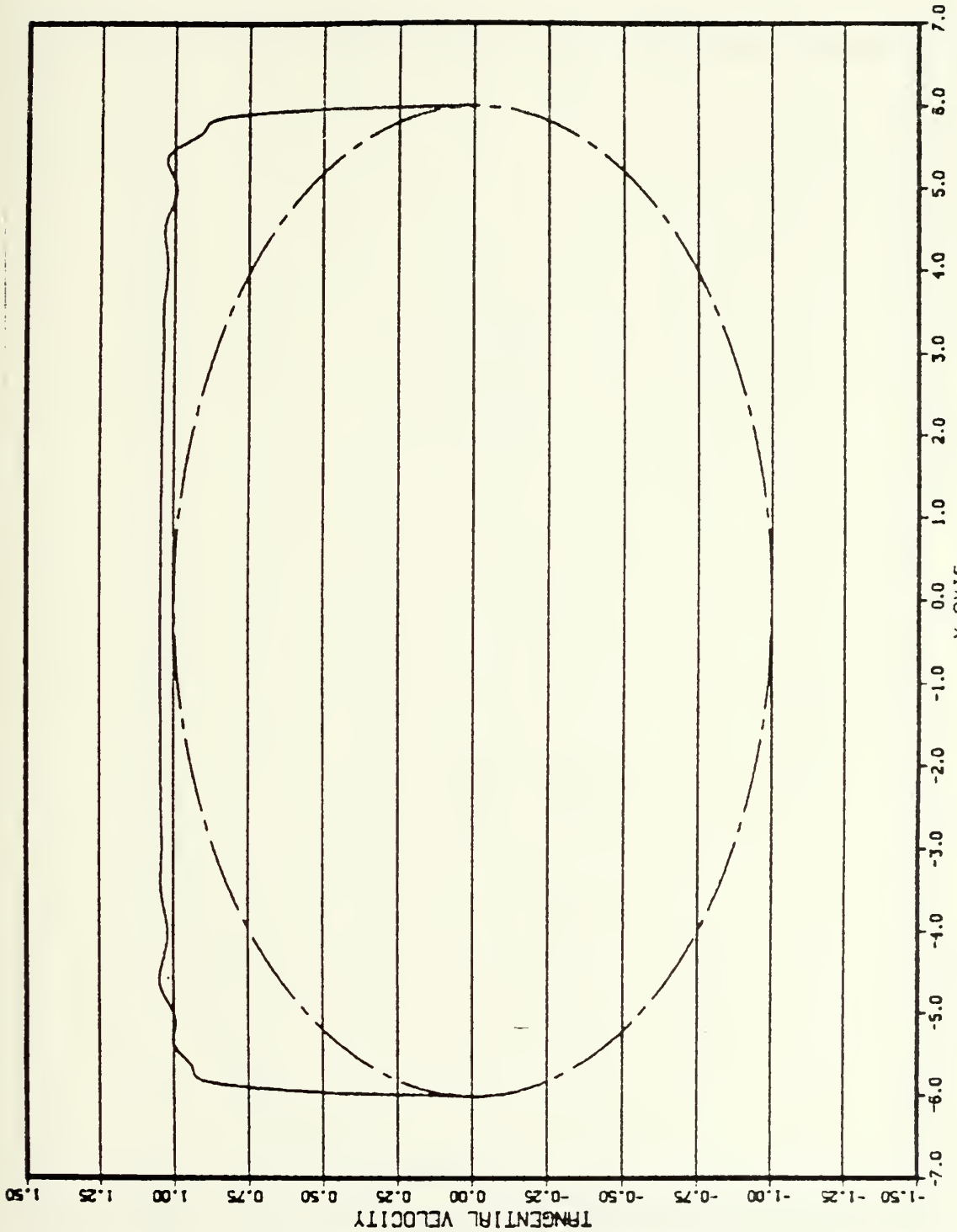


Figure 14b. Tangential Velocity for the Case of 20/22

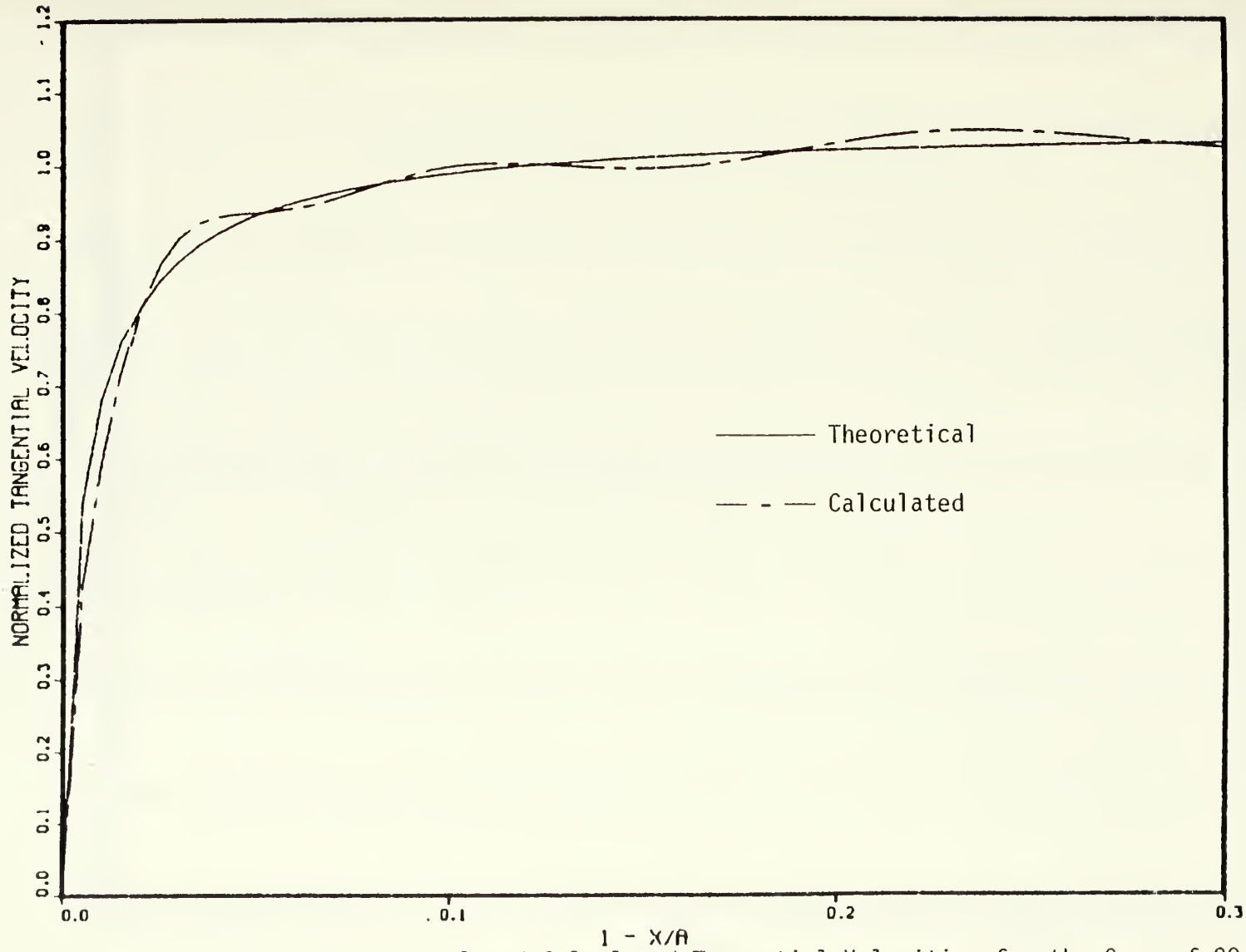


Figure 14c. Theoretical and Calculated Tangential Velocities for the Case of 20/22

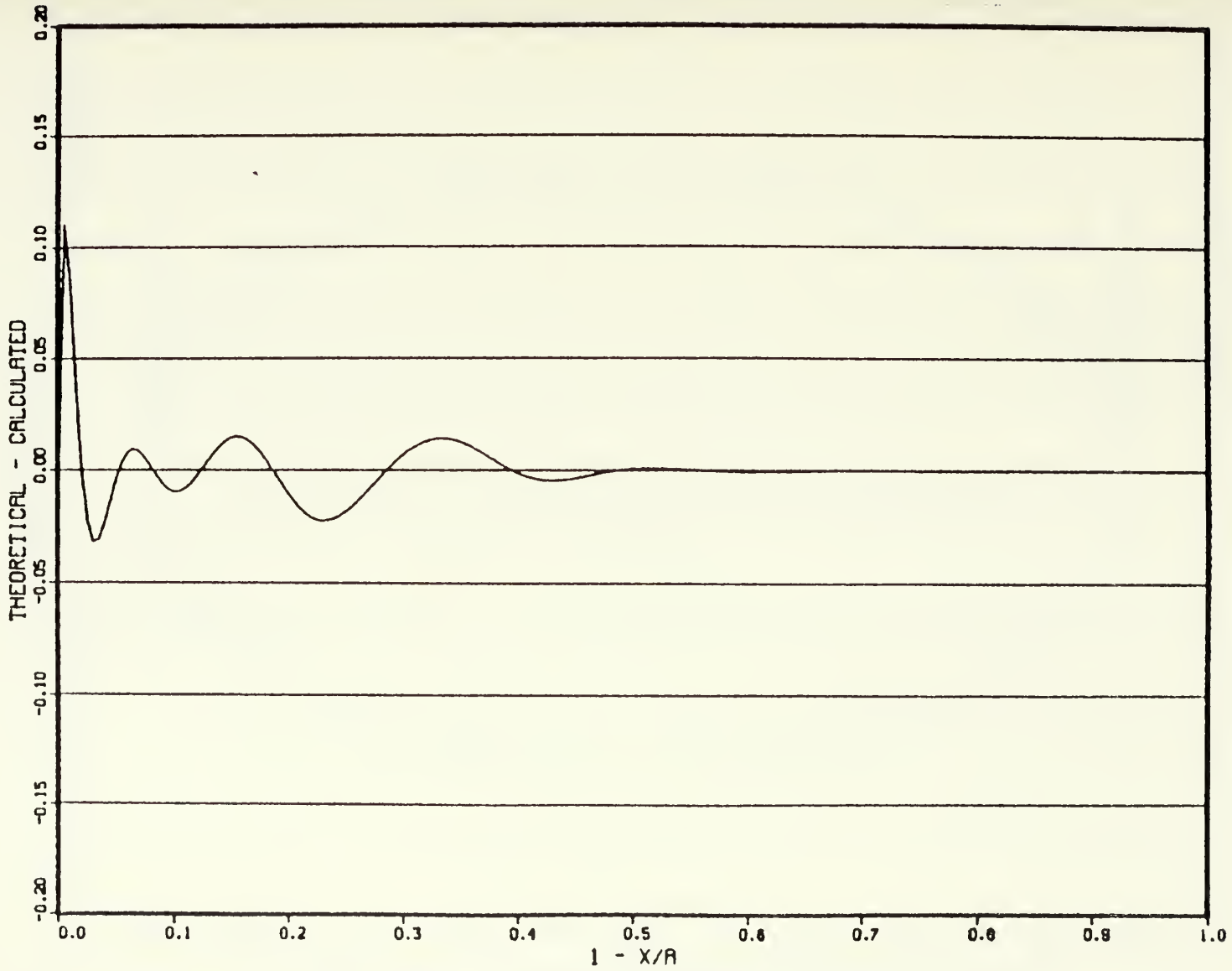


Figure 14d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 20/22

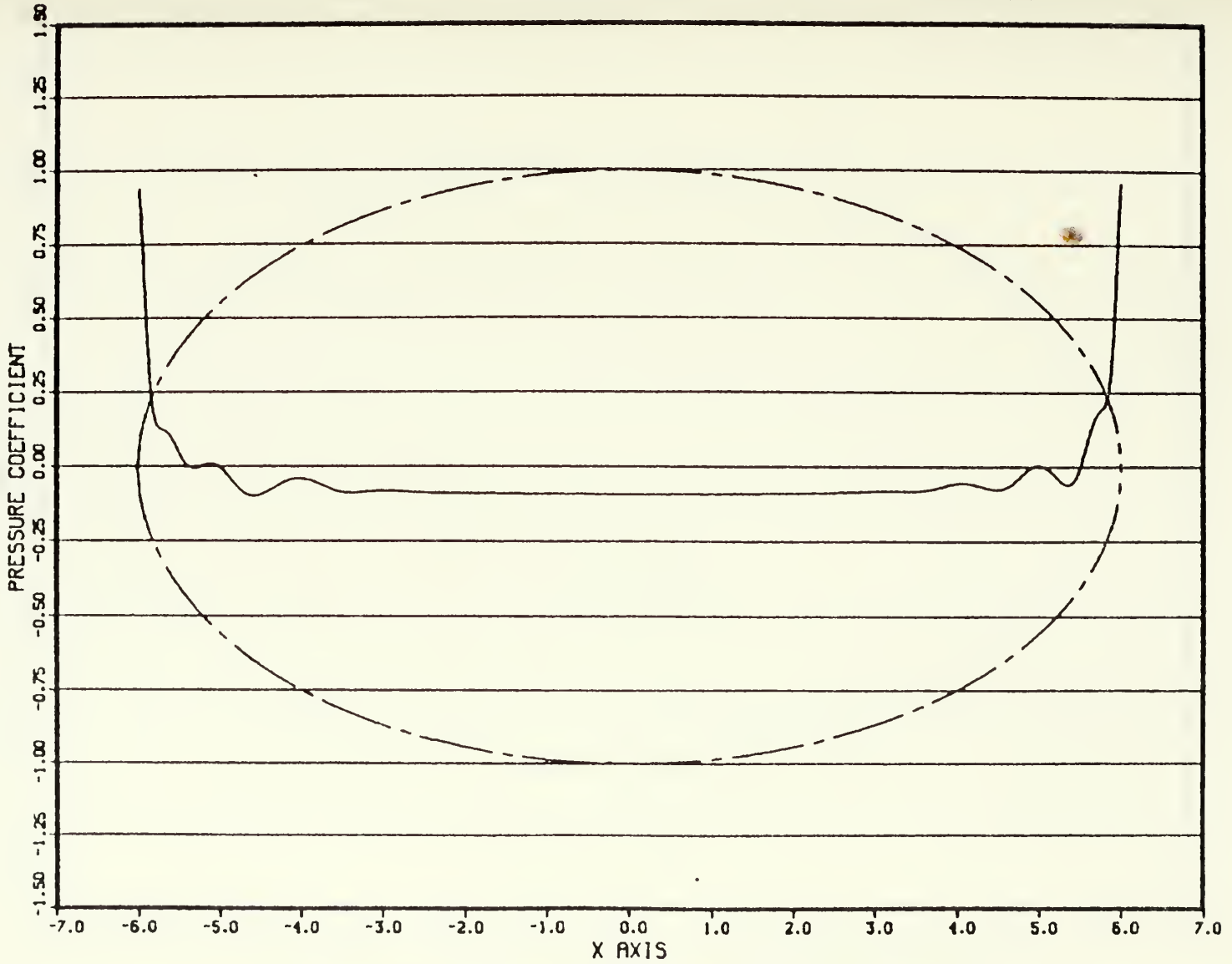


Figure 14e. Pressure Coefficient for the Case of 20/22

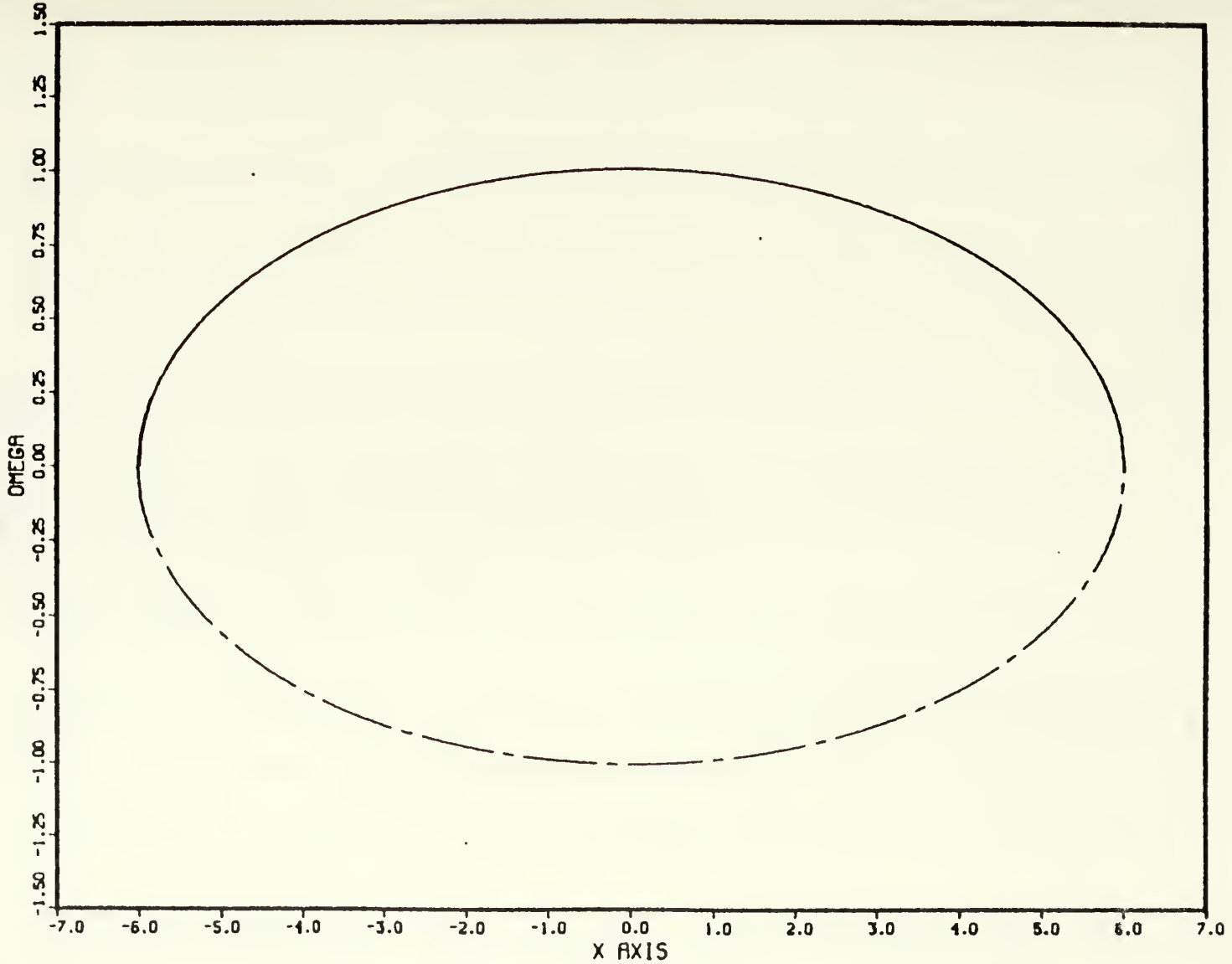


Figure 14f. Body Shape for the Case of 20/22

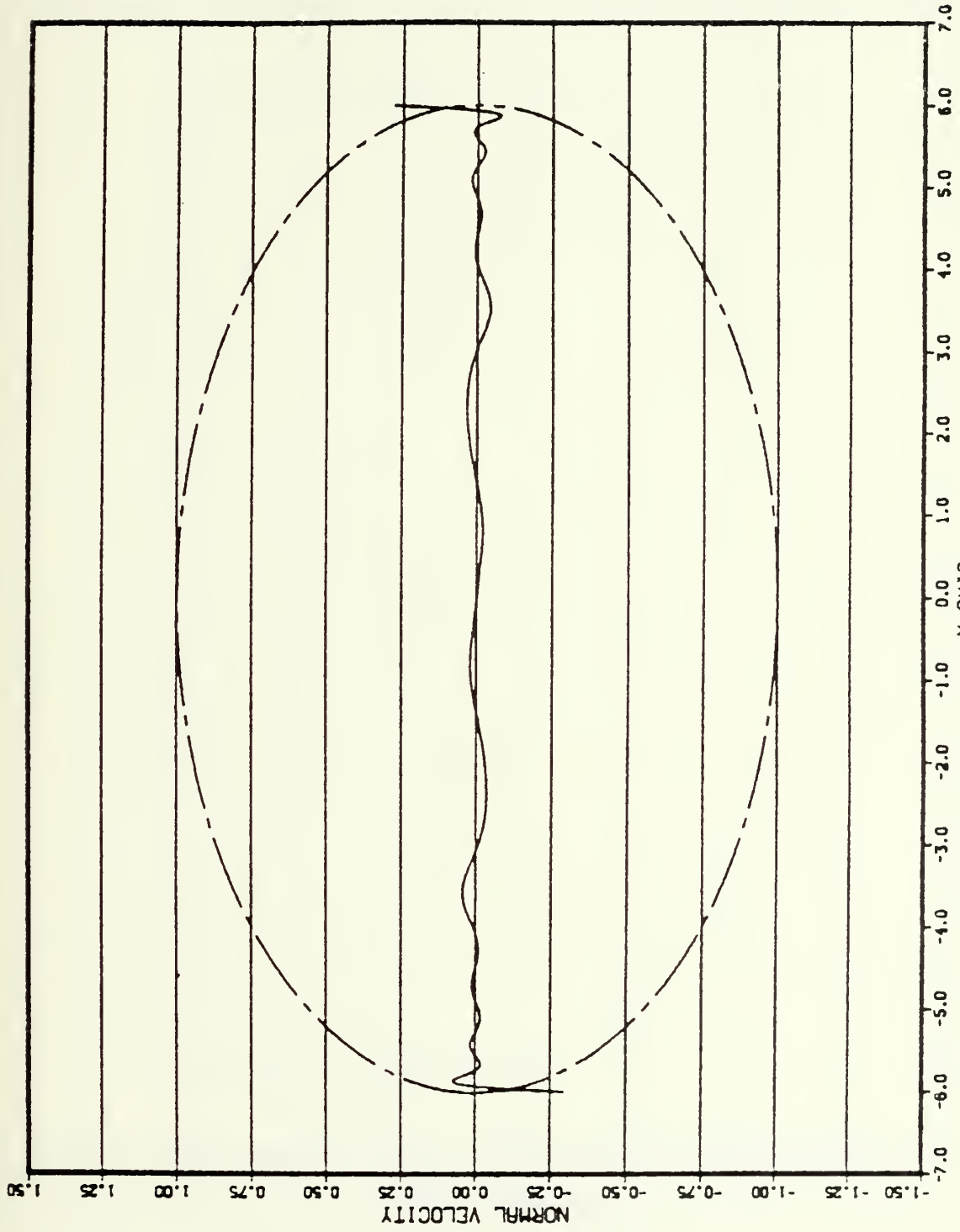


Figure 15a. Normal Velocity for the Case of 10/12

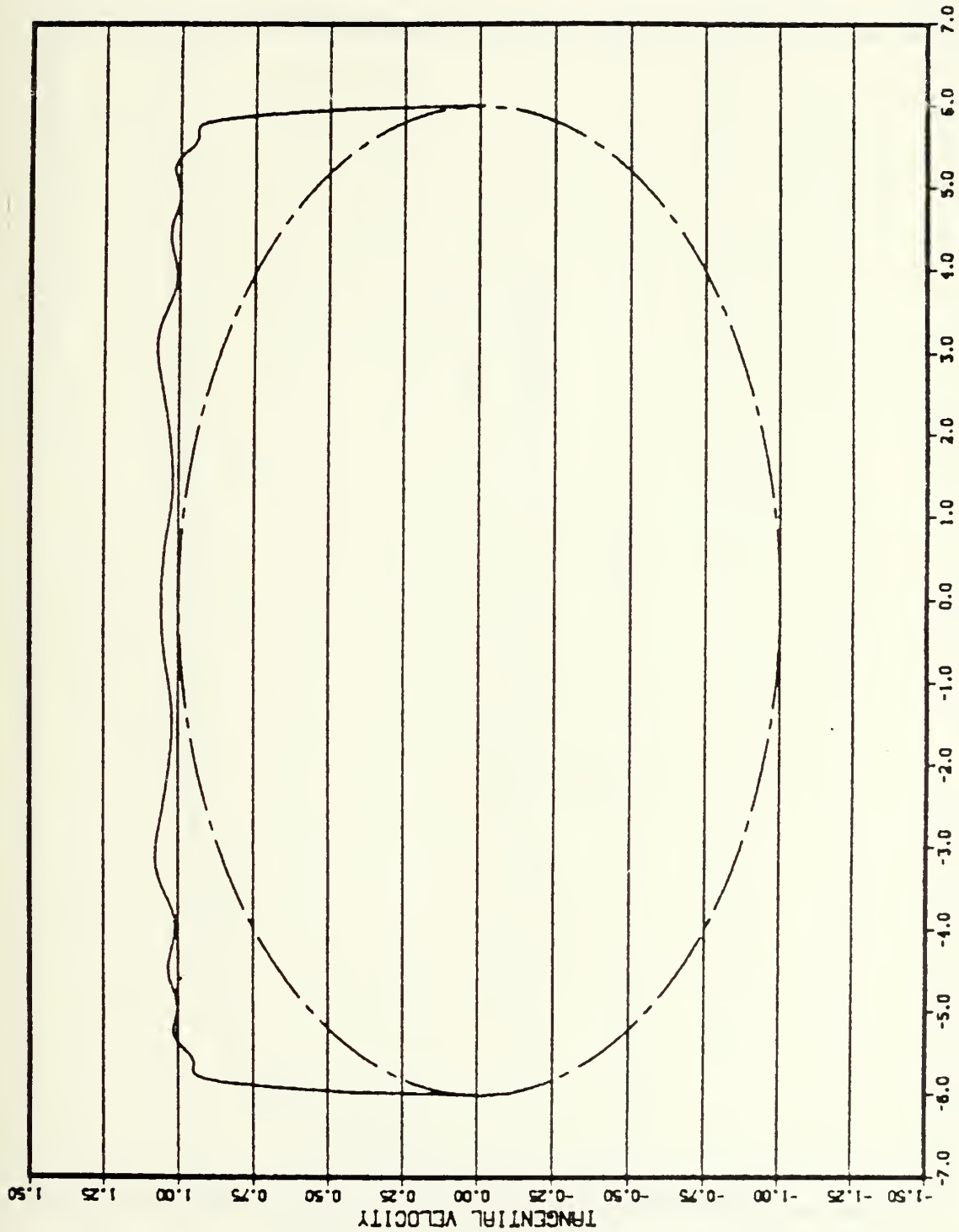


Figure 15b. Tangential Velocity for the Case of 10/12

Figure 15c. Theoretical and Calculated Tangential Velocities for the Case of 10/12

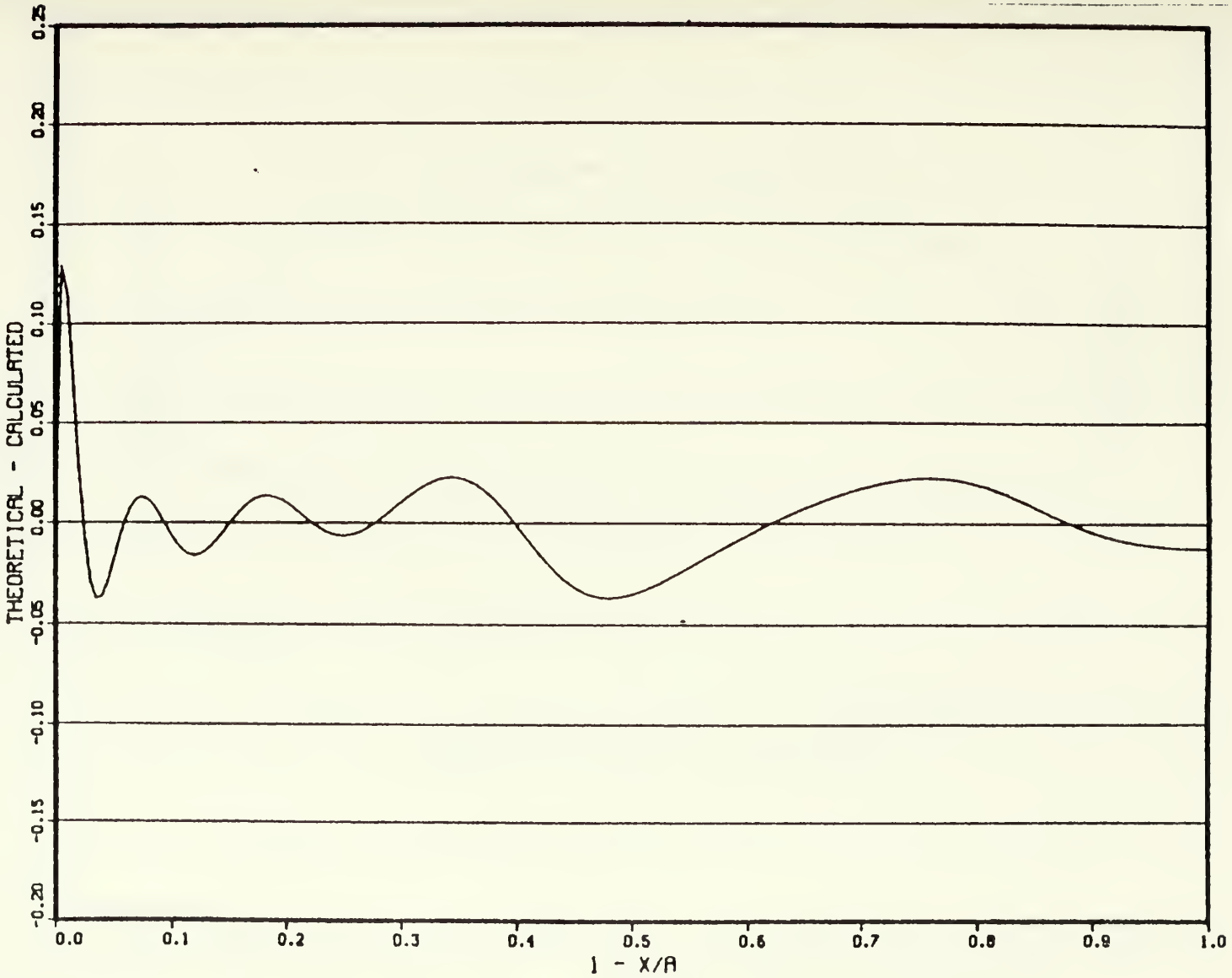


Figure 15d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 10/12

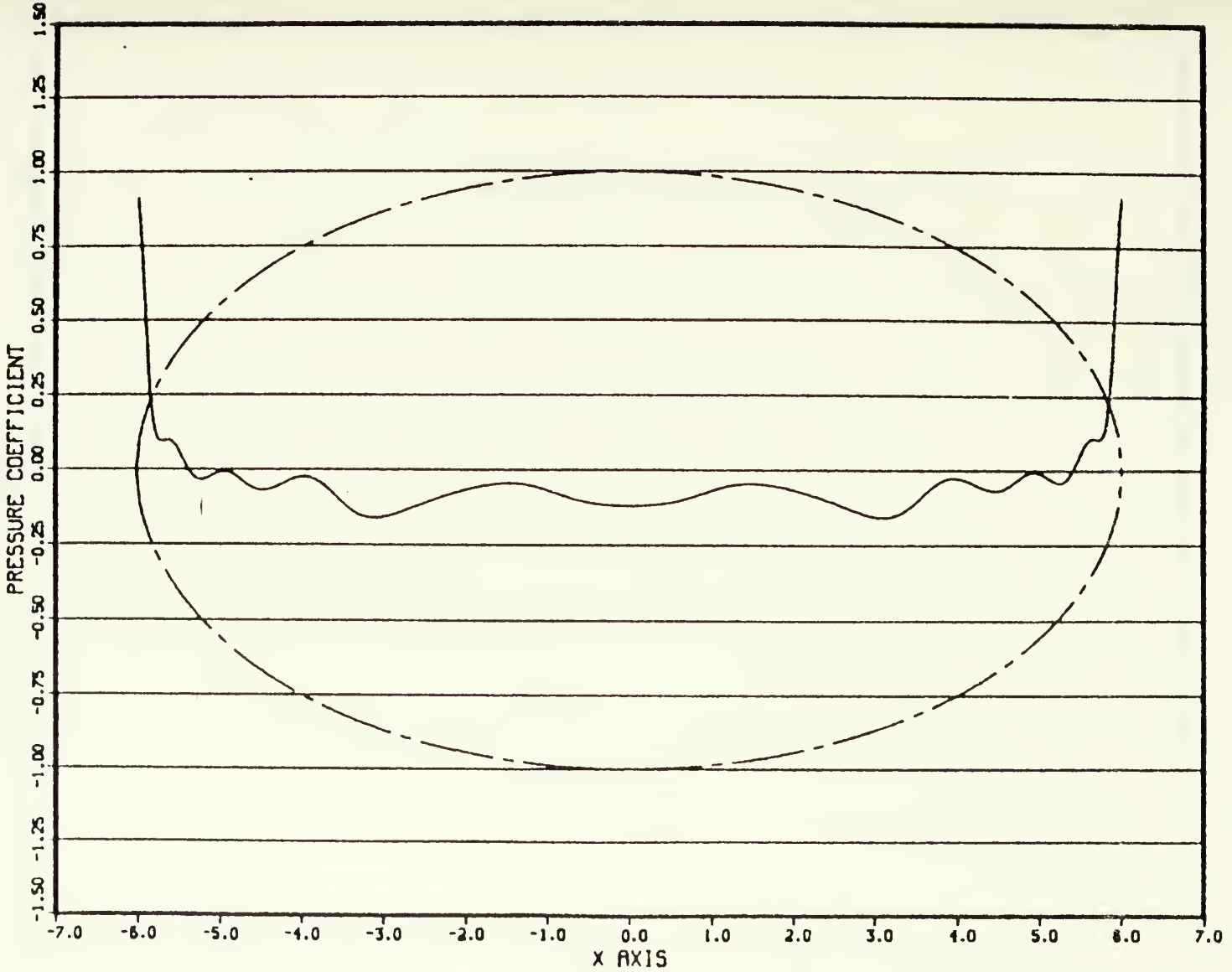


Figure 15e. Pressure Coefficient for the Case of 10/12

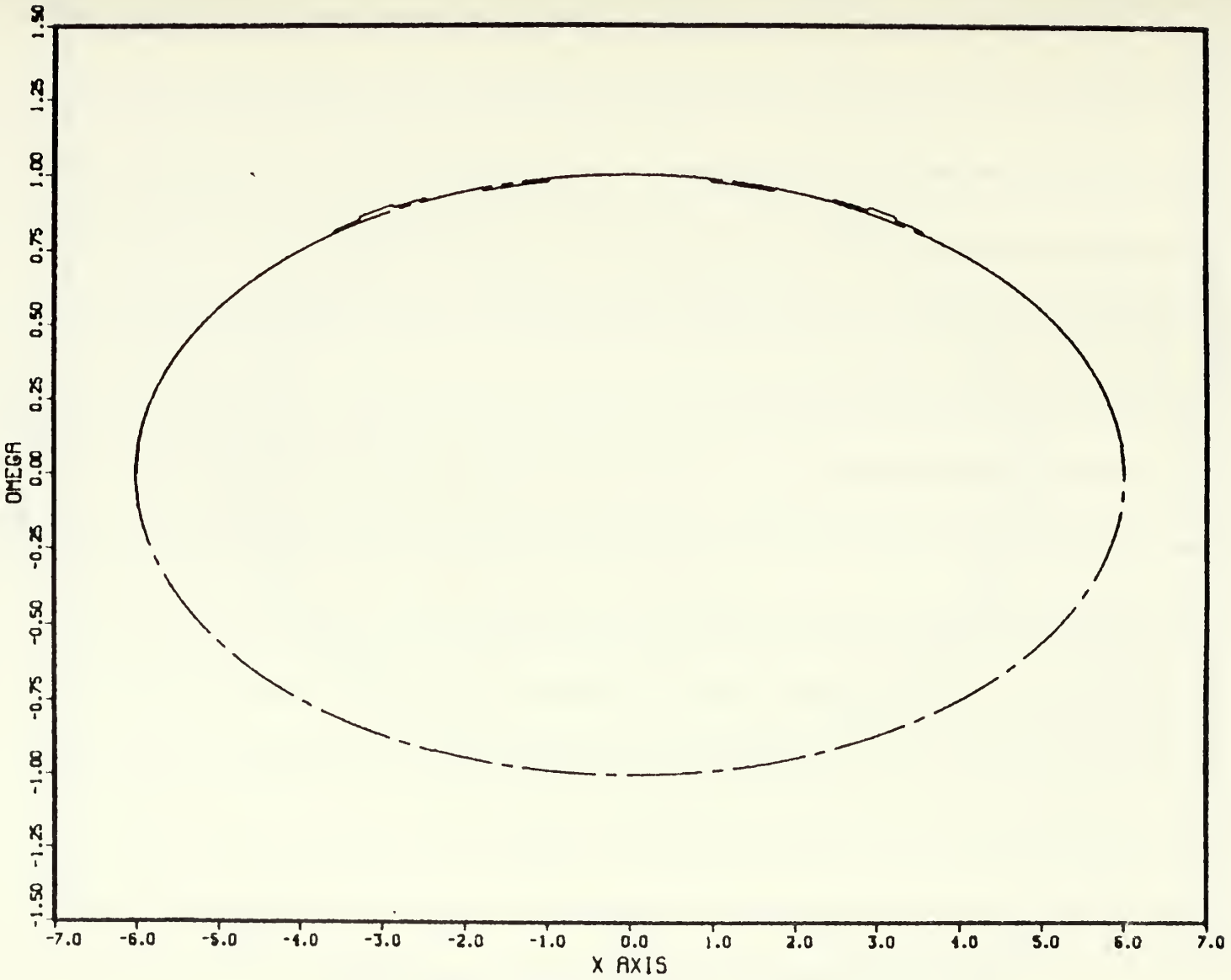


Figure 15f. Body Shape for the Case of 10/12

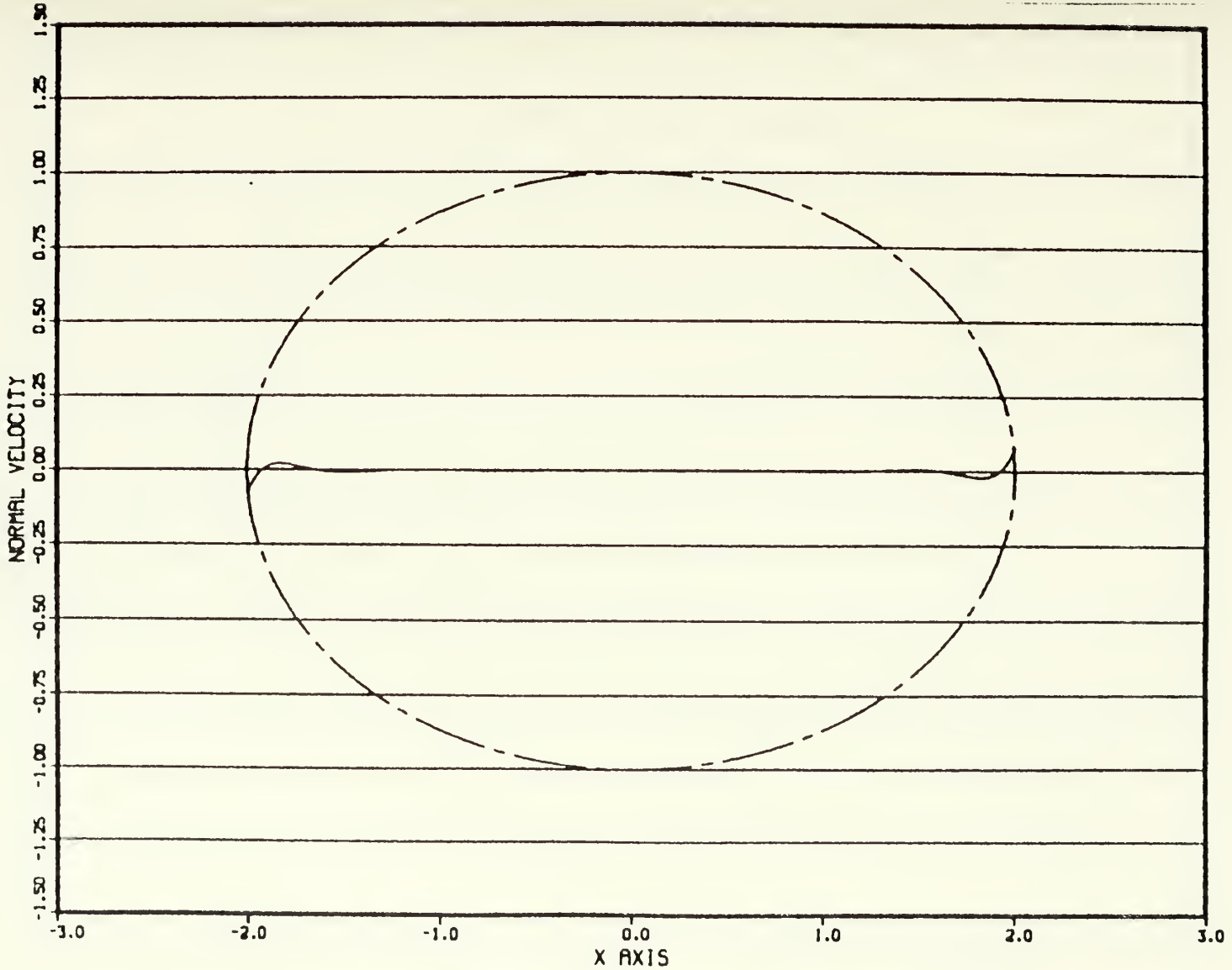


Figure 16a. Normal Velocity for the Case of 8/10

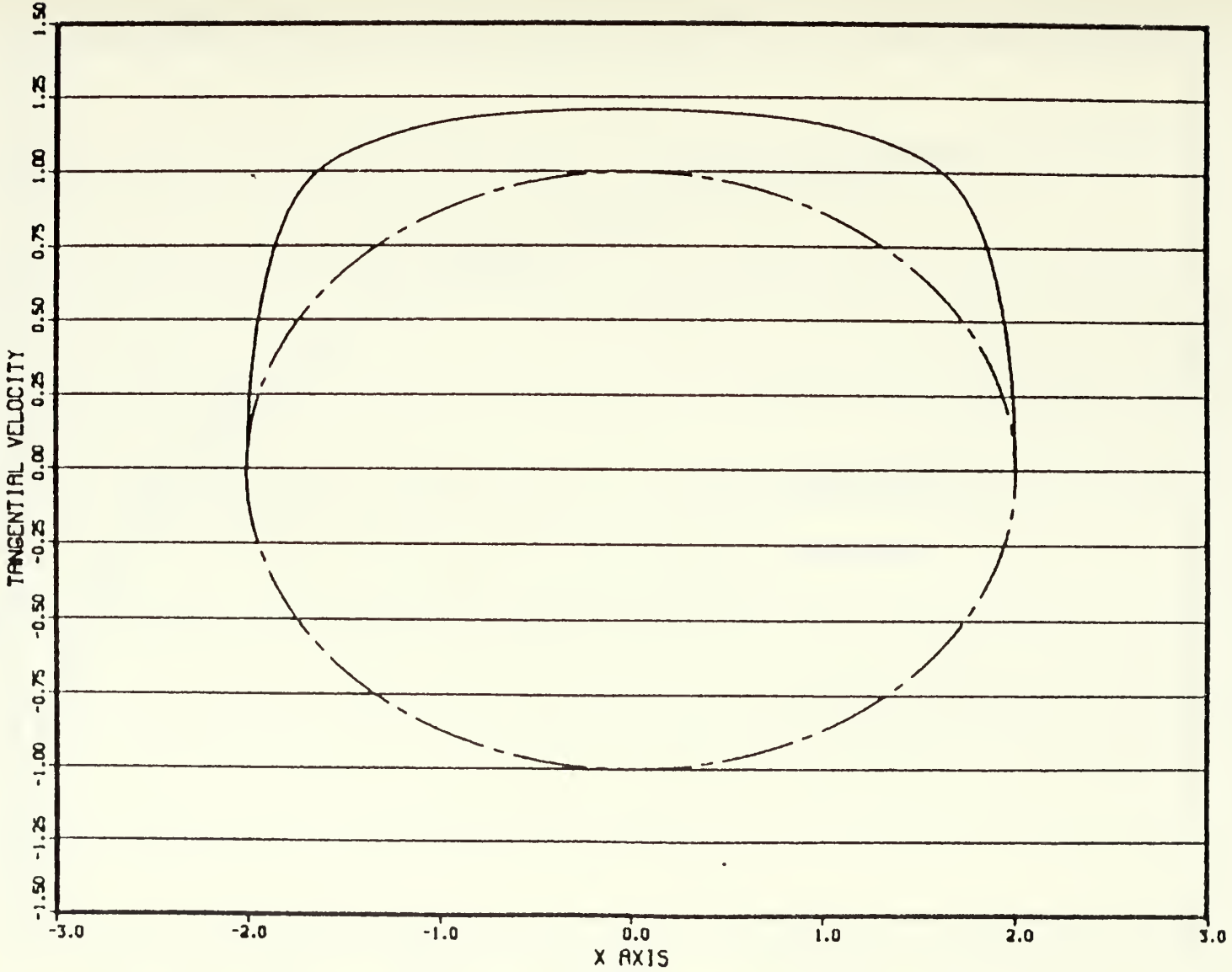


Figure 16b. Tangential Velocity for the Case of 8/10

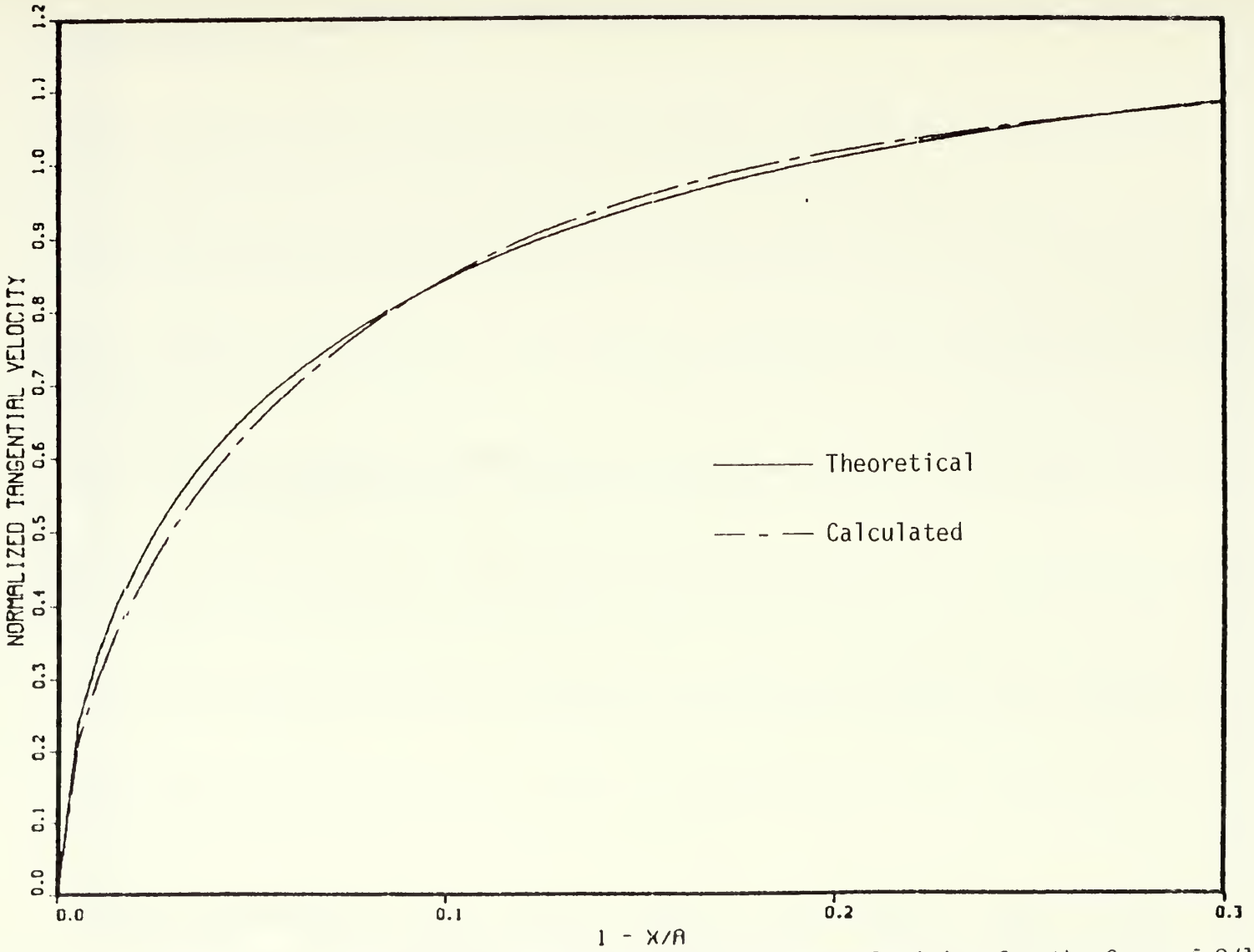


Figure 16c. Theoretical and Calculated Tangential Velocities for the Case of 8/10

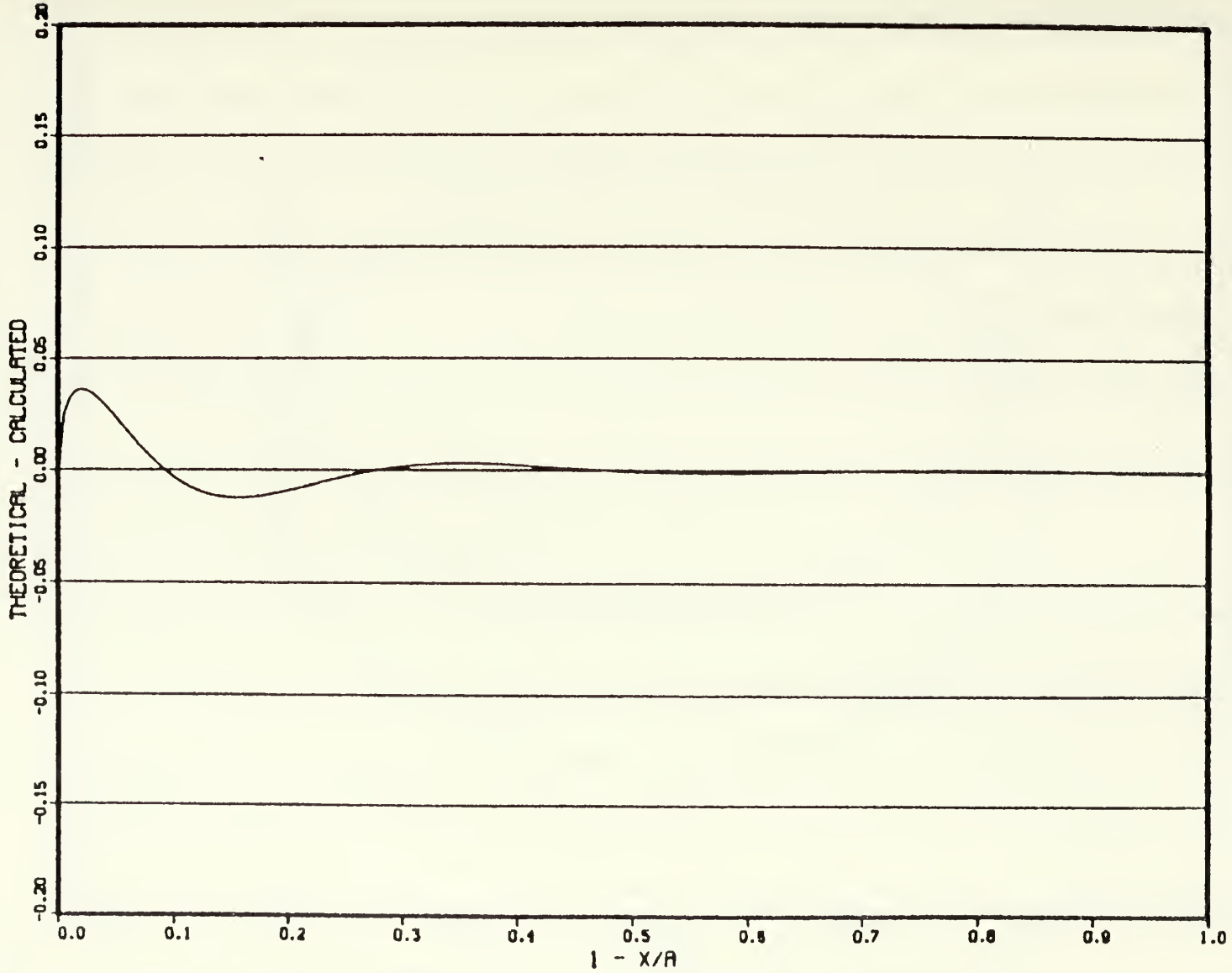


Figure 16d. Difference between the Theoretical and Calculated Tangential Velocities for the Case of 8/10

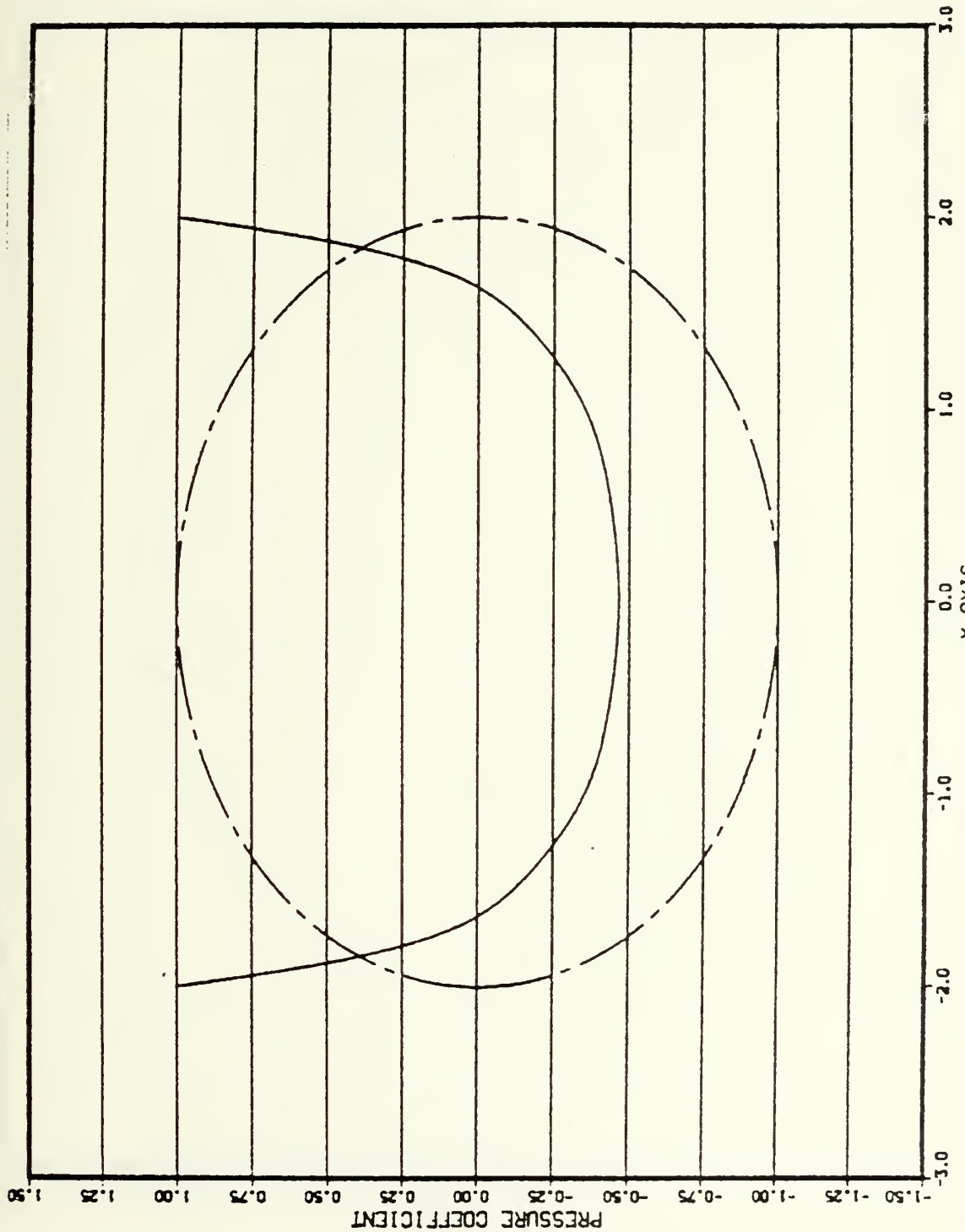


Figure 16e. Pressure Coefficient for the Case of 8/10

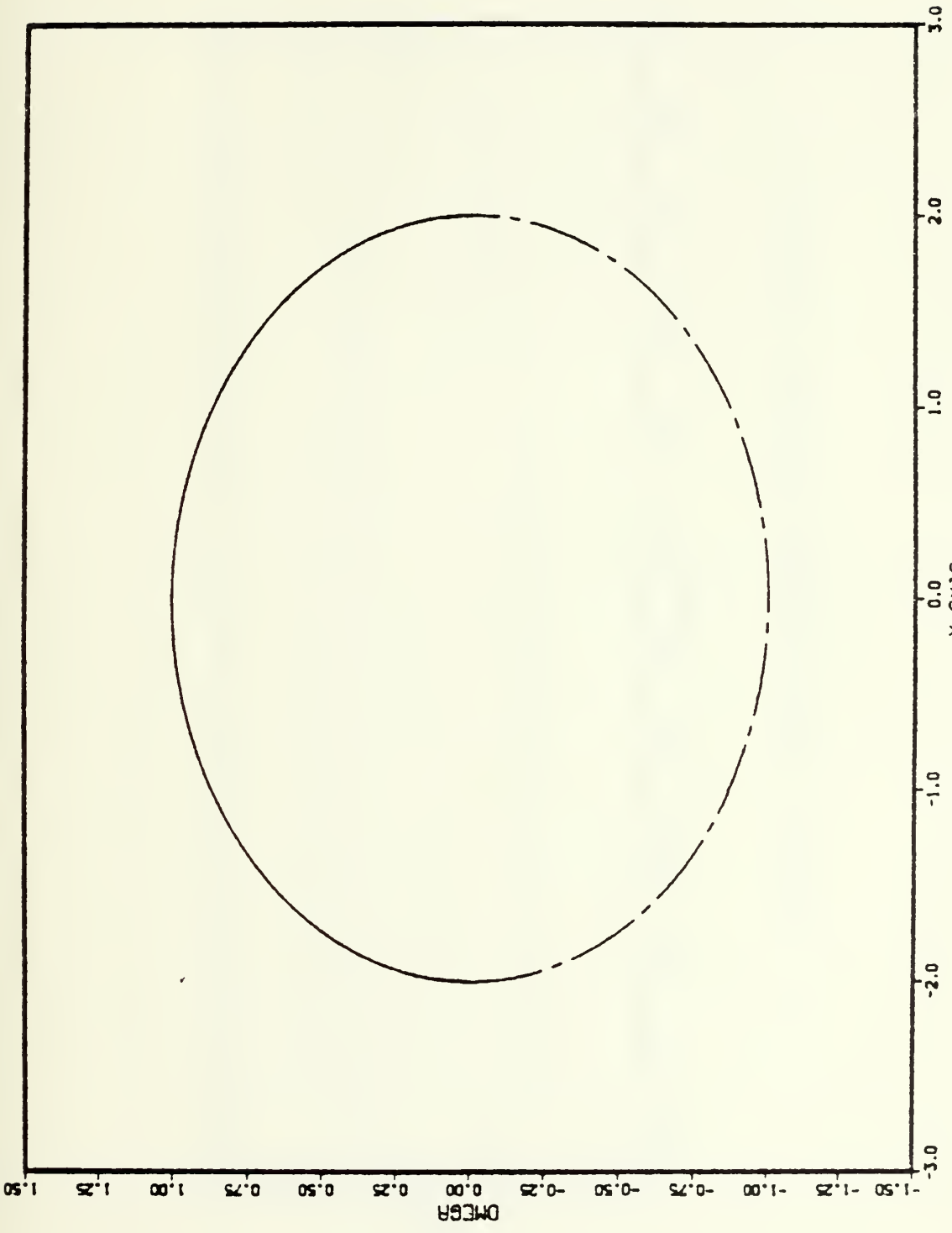


Figure 16f. Body Shape for the Case of 8/10

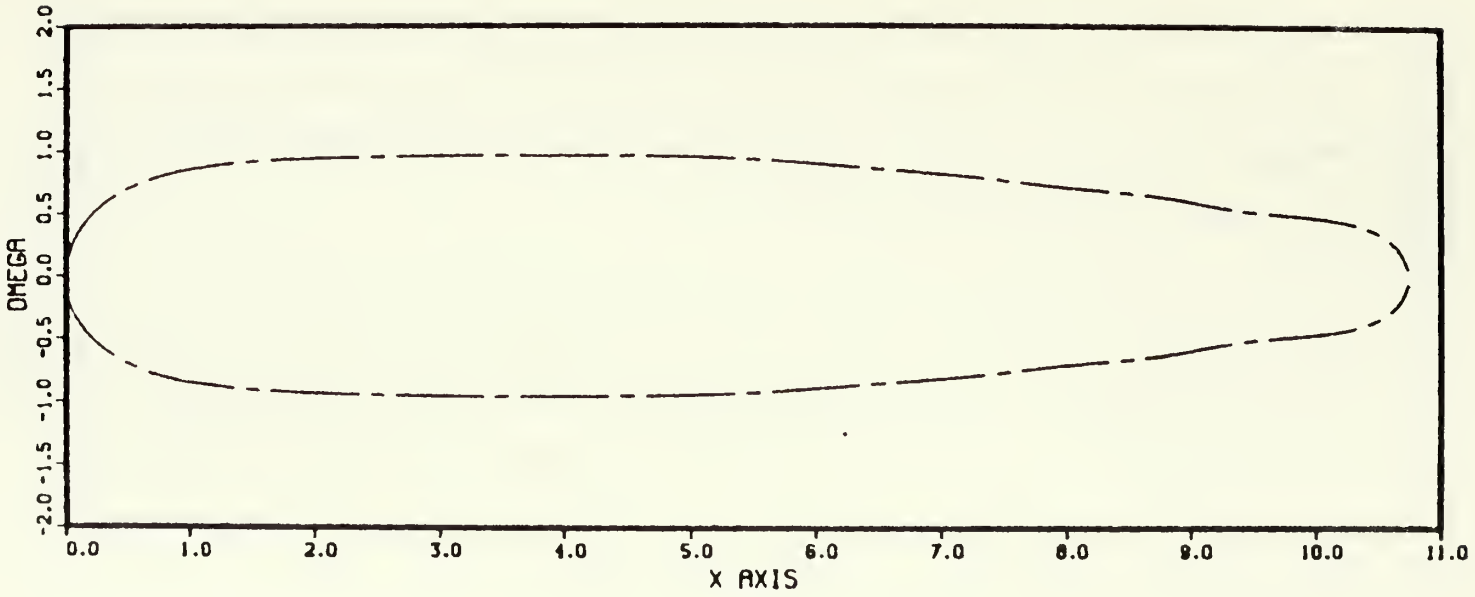


Figure 17. Tapered Axisymmetric Body Used in Analysis

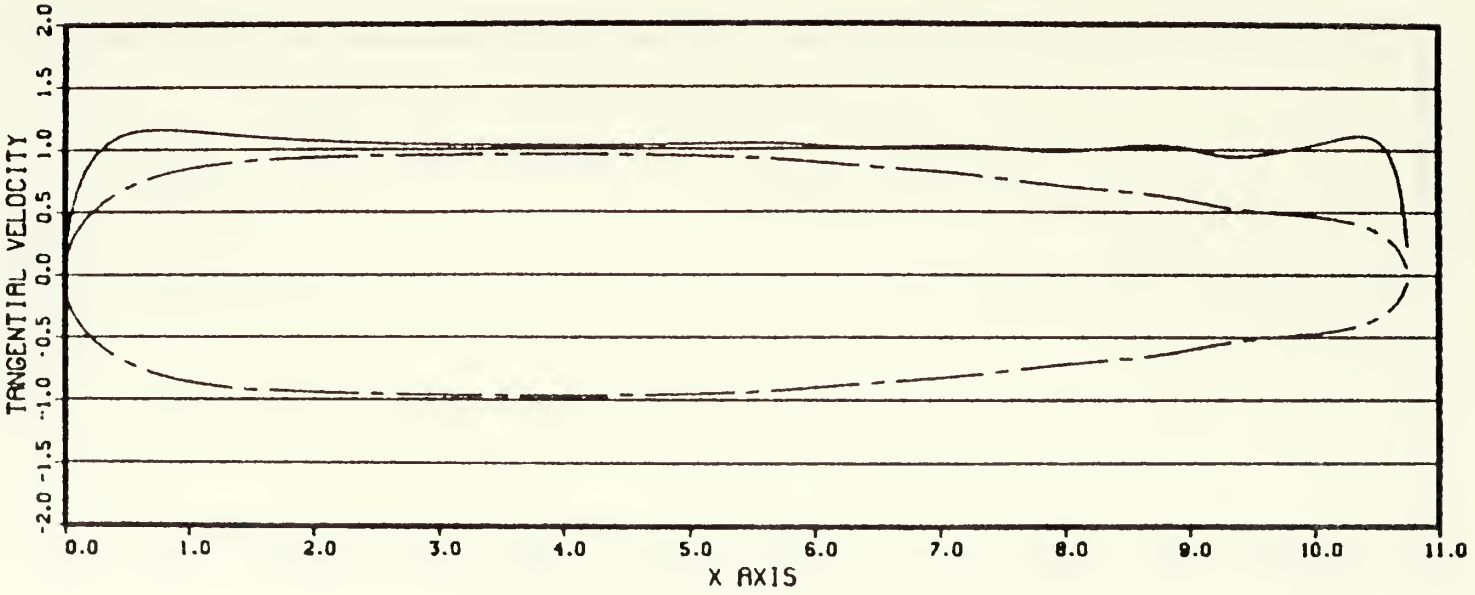


Figure 18. Theoretical Tangential Velocity for the Tapered Axisymmetric Body

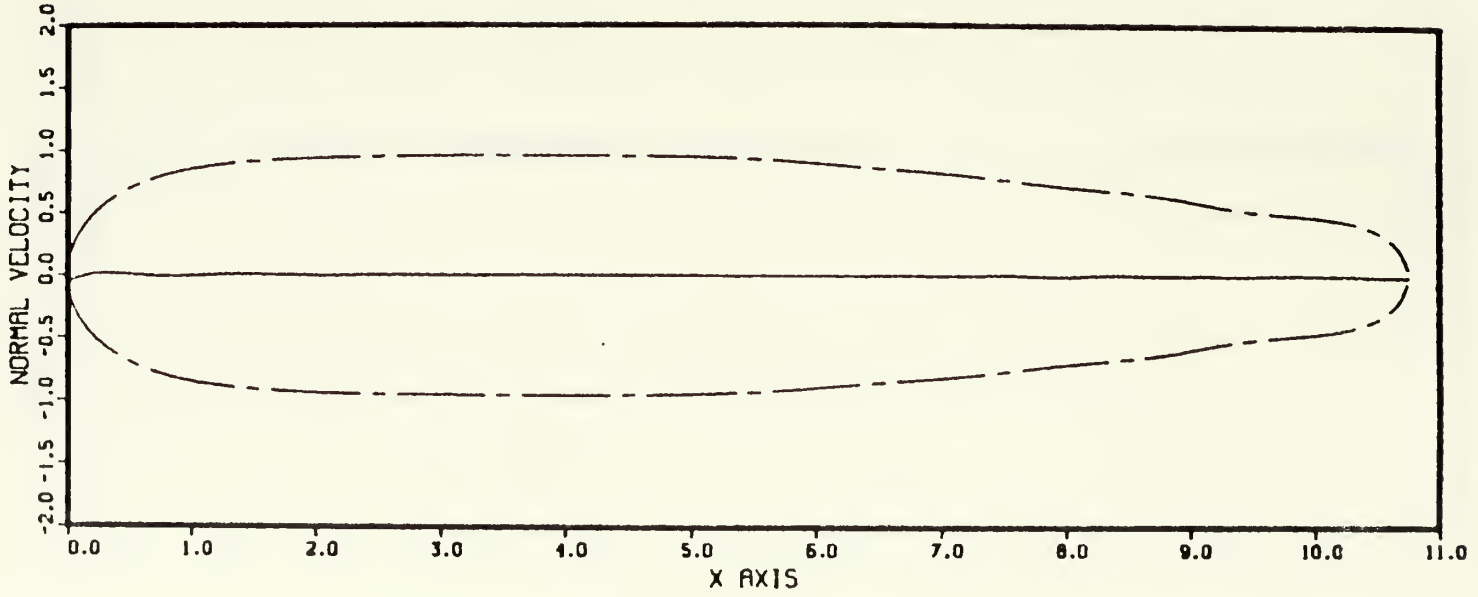


Figure 19a. Calculated Normal Velocity for the Tapered Axisymmetric Body

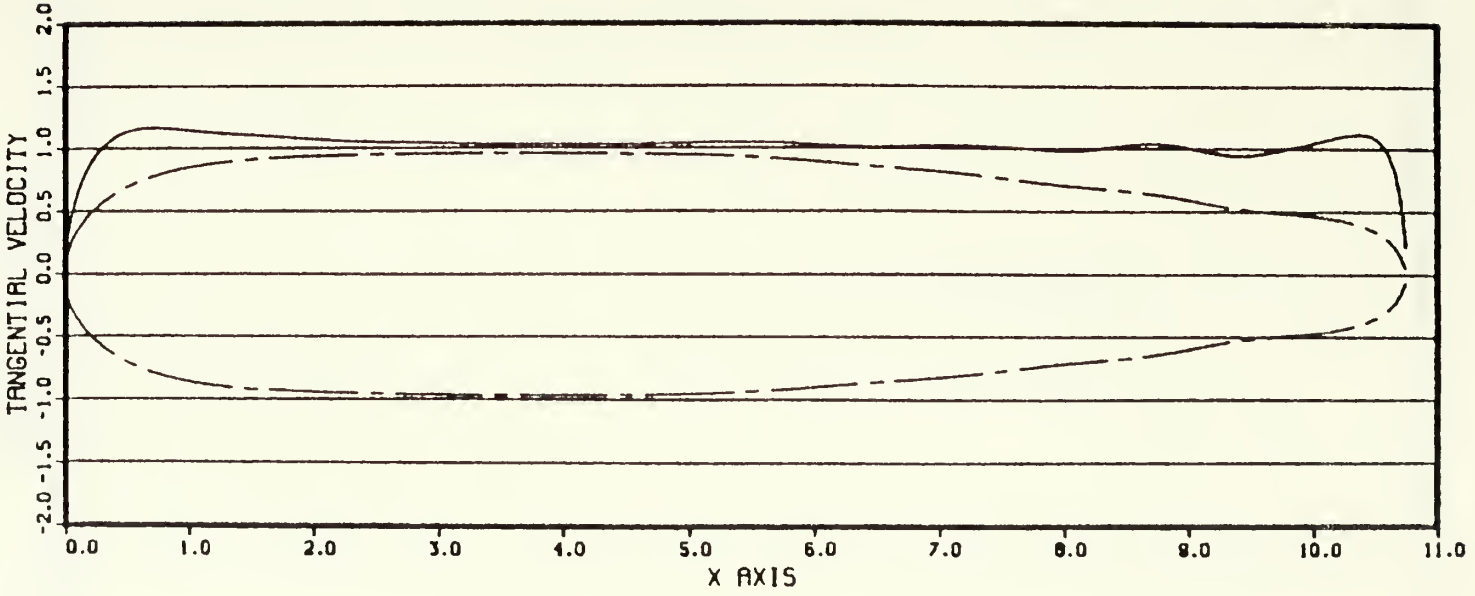


Figure 19b. Calculated Tangential Velocity for the Tapered Axisymmetric Body

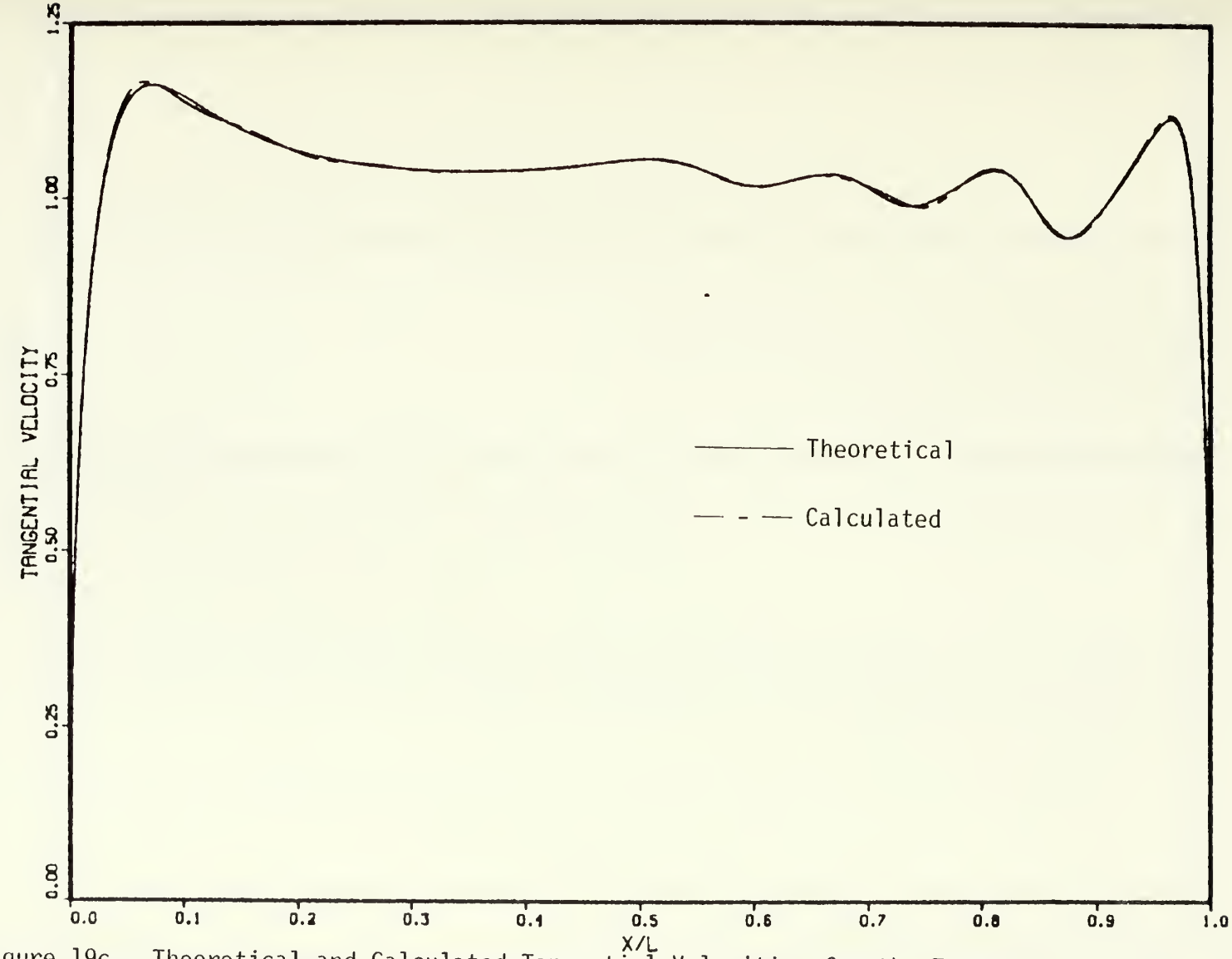


Figure 19c. Theoretical and Calculated Tangential Velocities for the Tapered Axisymmetric Body

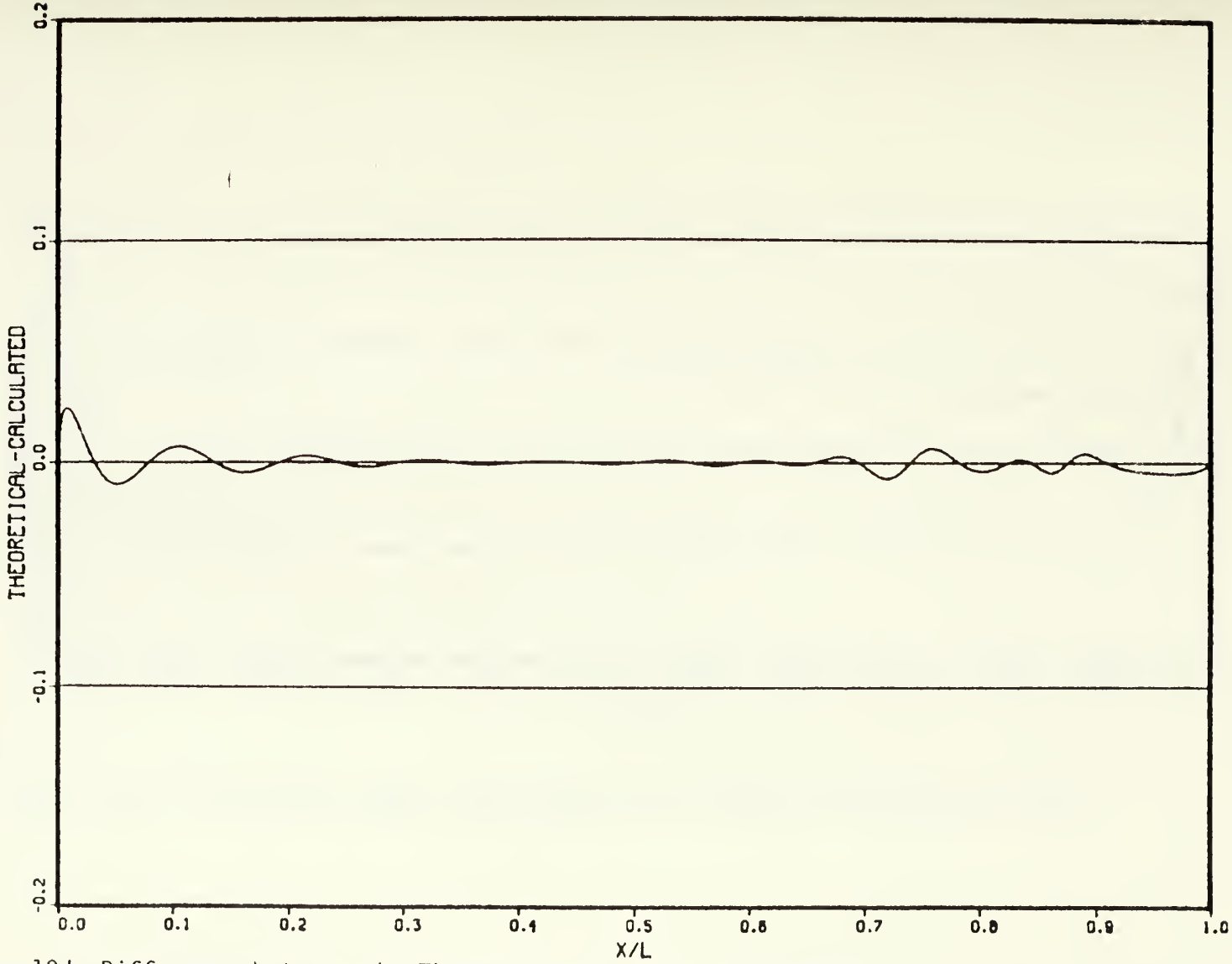


Figure 19d. Difference between the Theoretical and Calculated Tangential Velocities for the Tapered Axisymmetric Body

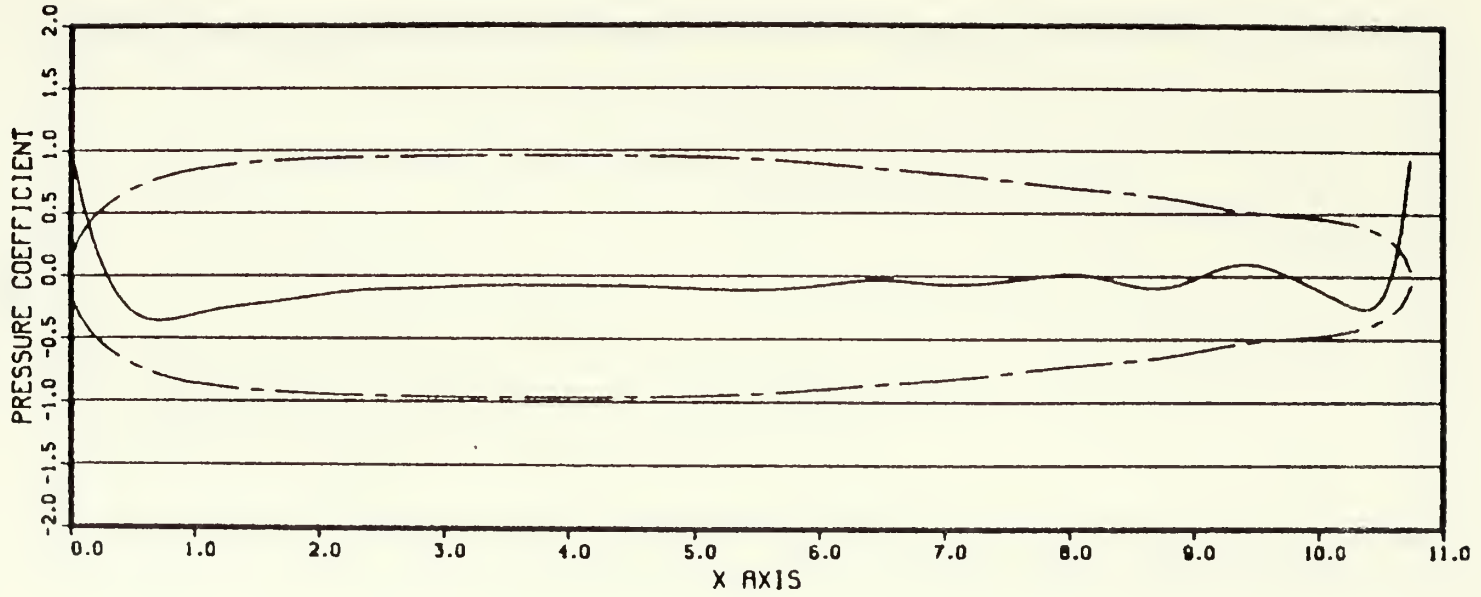


Figure 19e. Calculated Pressure Coefficient for the Tapered Axisymmetric Body

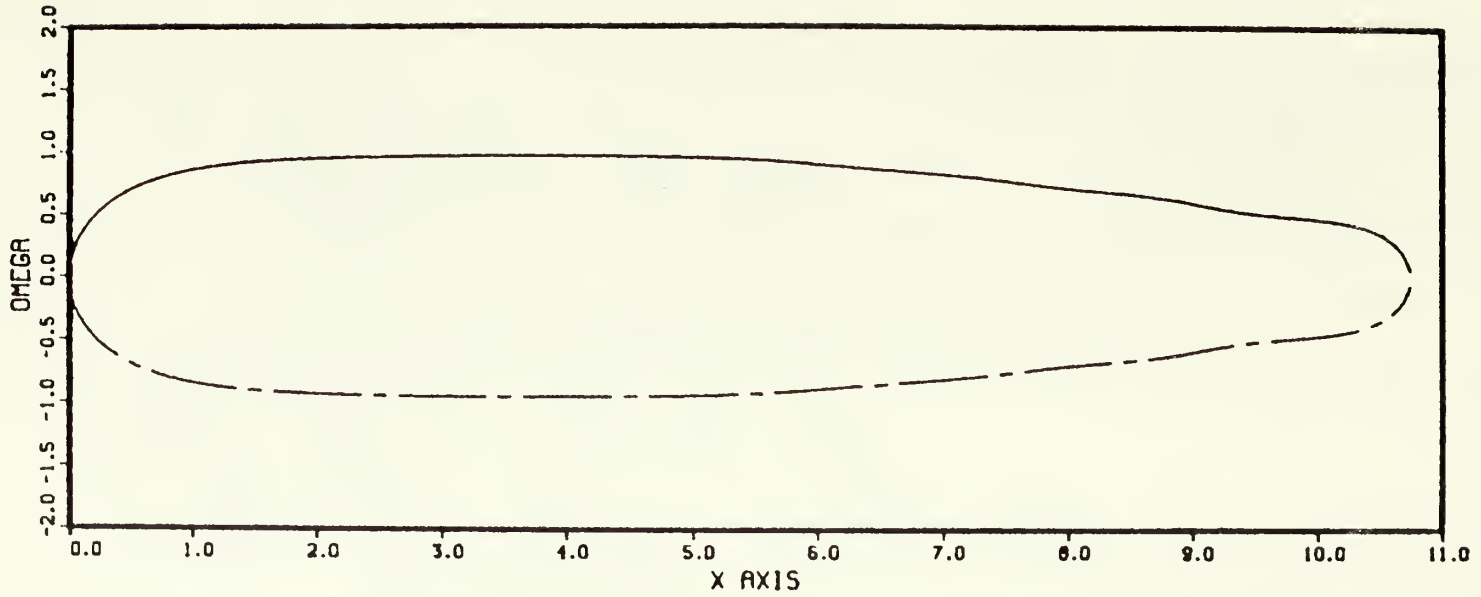


Figure 19f. Calculated Body Shape for the Tapered Axisymmetric Body

APPENDIX A: COMPUTER PROGRAM

```
* ************************************************************************
*   THE FOLLOWING PROGRAM IS USED TC MODEL POTENTIAL FLOW ABOUT ELLIP-  *
* SCIDS USING DISCRETE SINGULARITIES.  IT PERFORMS THE FOLLOWING:       *
*     1) USING THE GAUSS' LEAST SQUARES METHOD, SOLVES FOR THE          *
*        STRENGTHS OF THE SINGULARITIES.                                *
*     2) CALCULATES THE FLOW CHARACTERISTICS ABOUT THE BODY (I.E. V ,    *
*        V ,C )                                                         *
*     3) APPORTIONS CONTROL POINTS ALONG THE BODY SURFACE AS NEEDED.     *
*     4) PLOTS THE FLOW CHARACTERISTICS ABOUT THE BODY.                  *
* INPUT DATA REQUIRED IS DENOTED IN THE VARIABLE LIST THAT FOLLOWS.     *
* ************************* VARIABLES ********************************
* NSING - NUMBER OF SINGULARITIES (INPUT)                               *
* NPOINT - NUMBER OF CONTROL POINTS (INPUT)                             *
* A - HALF LENGTH OF MAJOR AXIS OF ELLIPSOID (INPUT)                    *
* B - HALF LENGTH OF MINOR AXIS OF ELLIPSOID (INPUT)                    *
* SPOSIT(I) - ARRAY CONTAINING THE POSITIONS OF THE SINGULARITIES ALONG*
*             THE X-AXIS (INPUT)
* CPOSIT(J) - ARRAY CONTAINING THE POSITIONS OF THE CONTROL POINTS      *
*             ALONG THE BODY SURFACE (INPUT)                            *
* RATIO - SLENDERNESS RATIO OF ELLIPSOID                                *
* OMEGA - RADIAL DISTANCE FROM MAJOR AXIS TO CONTROL POINT              *
```

```
* BB(J,1) - ARRAY OF TERMS IN STOKES STREAM FUNCTION EQUATION CONTRI-  *
*            BUTED BY FREE STREAM VELOCITY                             *
* R - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN CONTROL POINT*
* AA(J,I) - ARRAY OF TERMS IN STOKES STREAM FUNCTION EQUATION CONTRI-  *
*            BUTED BY THE SINGULARITIES                                *
* BB1(I,1) - ARRAY CONTAINING THE STRENGTHS OF THE SINGULARITIES       *
* SIGMAM - SUM OF THE STRENGTHS OF THE SINGULARITIES                   *
* X(K) - ARRAY OF EQUALLY SPACED POINTS ALONG THE BODY SURFACE         *
* OMEGA1 - RADIAL DISTANCE FROM MAJOR AXIS TO A GIVEN X(K) POINT       *
* R1 - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN X(K) POINT  *
* PSI - VALUE OF STOKES STREAM FUNCTION AT A GIVEN X(K) POINT          *
* U - VELOCITY COMPONENT IN THE X-DIRECTION AT A GIVEN X(K) POINT      *
* V - VELOCITY COMPONENT IN THE  -DIRECTION AT A GIVEN X(K) POINT      *
* ALPHA - ANGLE BETWEEN U AND TANGENTIAL VELOCITY COMPONENT AT A GIVEN *
*         X(K) POINT                                                   *
* VNORM(K) - ARRAY OF THE NORMAL VELOCITY COMPONENTS AT THE X(K) POINTS*
* AVNORM(K) - ARRAY OF ABSOLUTE VALUES OF VNORM(K)                     *
* SUMT - TOTAL SUM OF AVNORM(K) ALONG THE BODY SURFACE                 *
* VTANG - TANGENTIAL VELOCITY COMPONENT AT A GIVEN X(K) POINT          *
* CP - PRESSURE COEFFICIENT AT A GIVEN X(K) POINT                      *
* S - AVERAGE ROOT MEAN SQUARE VALUE OF THE NORMAL VELOCITIES          *
* SUM(K) - ARRAY OF SUM OF AVNORM(K) FOR A GIVEN INTERVAL ALONG THE    *
*          BODY SURFACE                                                *
* MCRECP(J) - ARRAY OF NUMBER OF ADDITIONAL CONTROL POINTS APPORTIONED *
*             TO A PARTICULAR INTERVAL ALONG THE BODY SURFACE           *
* DELTA(J) - ARRAY OF INTERVAL LENGTHS ALONG THE BODY SURFACE          *
* MAXD(J) - ARRAY OF MAXIMUM NUMBER OF CONTROL POINTS THAT CAN BE AP-  *
*           PORTIONED TO A GIVEN INTERVAL. (BASED ON A MAXIMUM DENSITY *
*           CRITERIA OF 15 CONTROL POINTS PER UNIT LENGTH)             *
* NUMCP(J) - NUMBER OF CONTROL POINTS ACTUALLY APPORTIONED TO A GIVEN  *
```

```
*            INTERVAL AFTER A COMPARISON OF MORECP(J) AND MAXD(J) FOR  *
*            THAT INTERVAL                                             *
* W(KK) - ARRAY OF RADIAL VALUES CORRESPONDING TO THE ZERO STREMLINE OF*
*         THE MODELED BODY                                             *
* sumnum - sum of all numcp(j)                                         *
* rii - radial distance from x-axis to a given x(kk) point             *
* si(kk) - value of stokes' stream function at a given x(kk) point     *
************************************************************************
      REAL*8 A,B,OMEGA,R,OMEGA1,STOR1,PSI,U,V,ALPHA,STORII,VTANG,CP
      REAL*8 STOR2,STOR3,R1,STOR4,S,XX,RII,AK,SIGMAM,SUMT,AVNORM(900)
      REAL*8 SPOSIT(50),CPOSIT(50),BB(50,1),AA(50,50),BB1(50,1),SUM(50)
      REAL*8 WKAREA(900),W(900),SI(900),AA1(50,50),X(900),VNORM(900)
      REAL*8 DELTA(50)
      REAL*4 XPLOT(900),VPLOT(900),CPLOT(900),OOPLOT(900),VTPLOT(900)
      REAL*4 CPPLOT(900),WPLOT(900)
      INTEGER I,J,K,NSING,NPOINT,KK,II,IFL,JFL,N1,N2,N3,N4,N5,MORECP(50)
      INTEGER N6,STOR7,STOR8,M,M1,M2,NUMCP(50)
      INTEGER Z1,Z2,Z3,COUNT,COUNT1,COUNT2,MAXD(50),SUMNUM
************************************************************************
* INPUT AXIS LENGTH, NUMBER AND POSITION OF SINGULARITIES AND CONTROL  *
* POINTS.                                                              *
************************************************************************
      READ(5,500) NSING,NPOINT,A,B
      READ(5,501) (SPOSIT(I),I=1,NSING)
      READ(5,501) (CPOSIT(J),J=1,NPOINT)
************************************************************************
* PASS DATA TO AUTOMATED DESIGN SYNTHESIS(ADS) OPTIMIZATION PROGRAM.   *
************************************************************************
      WRITE(3,300) A
      WRITE(3,300) B
```

```
      WRITE(3,301) NSING
      WRITE(3,300) (SPOSIT(I),I=1,NSING)
*************************************************************************
* CALCULATE SLENDERNESS RATIO. ECHO NUMBER AND POSITION OF CONTROL      *
* PCINTS AND SINGULARITIES TO OUTPUT FILE.                              *
*************************************************************************
      RATIO = A/B
      WRITE(6,600) RATIC
      DO 10 I=1,NSING
            WRITE(6,601) I,SPOSIT(I)
  10  CONTINUE
      DO 20 J=1,NPOINT
            WRITE(6,602) J,CPOSIT(J)
  20  CONTINUE
*************************************************************************
* SOLVE FOR STRENGTHS OF SINGULARITES FROM EQUATIONS OF THE STREAM      *
* FUNCTICN AT EACH CONTFCL POINT.                                       *
*************************************************************************
      DO 30 J=1,NPOINT
            OMEGA = DSQRT(1.D0-(CPOSIT(J)/A)**2)
            BB(J,1) = 0.5D0*OMEGA**2
            DO 40 I=1,NSING
                  R = DSQRT((((CPOSIT(J)-SPOSIT(I))/A)**2) + (OMEGA**2*
     *                  ((B/A)**2)))
                  AA(J,I) =(((CPOSIT(J)-SPOSIT(I))/A)/R)
  40        CONTINUE
  30  CONTINUE
*************************************************************************
* INITIALIZE INPUT DATA NECESSARY FOR CALLING LIBRARY SUBROUTINES       *
* VMULFM AND LEQT2F.                                                    *
```

```
*************************************************************************
      N = 1
      IA = 50
      IDGT = 10
      CALL VMULFM(AA,AA,NPOINT,NSING,NSING,IA,IA,AA1,IA,IER)
      CALL VMULFM(AA,BB,NPOINT,NSING,N,IA,IA,BB1,IA,IER)
      CALL LEQT2F(AA1,N,NSING,IA,BB1,IDGT,WKAREA,IER)
      SIGMAM = 0.D0
*************************************************************************
* WRITE VALUES OF THE STRENGTHS OF THE SINGULARITIES IN OUTPUT FILE AND*
* PASS TO ADS OPTIMIZATION PROGRAM.                                     *
*************************************************************************
      DO 50 I=1,NSING
            WRITE(6,603) I,BB1(I,1)
            WRITE(3,300) BB1(I,1)
            SIGMAM = SIGMAM + BB1(I,1)
 50   CONTINUE
      WRITE(6,608)  SIGMAM
      WRITE(6,604)
      WRITE(6,605)
*************************************************************************
* CALCULATE THE STREAM FUNCTION AT A MULTITUDE OF EQUALLY SPACED POINTS*
* ALONG THE BODY. ESTABLISH PLOTTING ARRAYS FOR AXISYMMETRIC BODY BEING*
* MODELED.                                                              *
*************************************************************************
      STOR4 = 0.D0
      SUMT = 0.D0
      DO 60 K = 1,401
            X(K) = (-A - (2.D0*A/400.D0)) + (2.D0*A/400.D0)*DFLOAT(K)
            XPLOT(K) = SNGL(X(K))
```

```
            OMEGA1 = DSQRT(1.D0-(X(K)/A)**2)
            OPLOT(K) = SNGL(OMEGA1)
            OOPLOT(K) = -OPLOT(K)
            STOR1 = 0.D0
            DO 70 I =1,NSING
                  R1 = DSQRT(((X(K)-SPOSIT(I))/A)**2)+(OMEGA1**2*(B/A)*
     *                 *2))
                  STOR1 = STOR1 + BE1(I,1)*((X(K)-SPOSIT(I))/A)/R1
  70        CONTINUE
            PSI = (0.5D0*OMEGA1**2) - STOR1
*************************************************************
* CALCULATE NORMAL VELOCITY, TANGENTIAL VELOCITY, AND THE PRESSURE CO- *
* EFFICIENT AT EACH  (X,OMEGA)  POINT ON THE BODY SURFACE.  ESTABLISH  *
* PLOTTING ARRAYS FOR SAME.                                            *
*************************************************************
            STOR2 = 0.D0
            STOR3 = 0.D0
            DO 80 I=1,NSING
                  R1 = DSQRT(((X(K)-SPOSIT(I))/A)**2)+(OMEGA1**2*(B/A)*
     *                 *2))
                  STOR2 = STOR2 + BE1(I,1)*(((X(K)-SPOSIT(I))/A)/R1**3)*
     *                    (B/A)**2
                  STOR3 = STOR3 + BE1(I,1)*((OMEGA1/R1**3)*(B/A)**3)
  80        CONTINUE
            U = 1.D0 + STOR2
            V = STOR3
            ALPHA = DATAN(-(X(K)/A)/OMEGA1*(B/A))
            VNORM(K) = V*DCOS(ALPHA) - U*DSIN(ALPHA)
            STOR4 = STOR4 + VNORM(K)**2
            AVNORM(K) = DABS(VNORM(K))
```

```
            SUMT = SUMT + AVNORM(K)
            VPLOT(K) = SNGL(VNORM(K))
            VTANG = U*DCOS(ALPHA) + V*DSIN(ALPHA)
            VTPLOT(K) = SNGL(VTANG)
            CP = 1.D0 - (U**2 + V**2)
            CPPLOT(K) = SNGL(CP)
            WRITE(6,606) X(K),PSI,VNORM(K),VTANG,CP,U,V
            WRITE(4,400) X(K),VTANG
  60  CONTINUE
*************************************************************
* CALCULATE THE AVERAGE ROOT MEAN SQUARE VALUE OF THE NORMAL VELOCITIES*
*************************************************************
      S = DSQRT(STOR4)/401.D0
      WRITE(6,606) S
*************************************************************
* DIVIDE THE BODY INTO INTERVALS BOUNDED BY SUCCESSIVE SINGULARITIES.  *
* APPORTION TO EACH INTERVAL THE NUMBER OF CONTROL POINTS REQUIRED     *
* BASED ON THE RATIO OF THE SUM OF THE ABSOLUTE VALUES OF THE NORMAL   *
* VELOCITIES OF THAT INTERVAL TO THE TOTAL SUM OF THE ABSOLUTE VALUES  *
* OF THE NORMAL VELOCITIES FOR THE WHOLE BODY.                         *
*************************************************************
      N1 = NSING + 2
      DO 110 K=1,N1
            SUM(K) = 0.D0
 110  CONTINUE
      N2 = NSING/2
      N3 = N2 + 1
      N4 = N2 - 1
      N5 = NSING - 1
      DO 120 K=1,401
```

```
            IF((X(K) .GT. -A) .AND. (X(K) .LT. SPOSIT(1))) GO TO 121
            DO 130 I=1,N4
                   IF((X(K) .GT. SPOSIT(L)) .AND. (X(K) .LT. SPOSIT(L
     *             +1))) GO TO 122
130         CONTINUE
            IF((X(K) .GT. SPOSIT(N2)) .AND. (X(K) .LT. 0.D0)) GO TO
     *      123
            IF((X(K) .GT. 0.D0) .AND. (X(K) .LT. SPOSIT(N3))) GO TO
     *      124
            DO 140 J= N3,N5
                   IF((X(K) .GT. SPOSIT(J)) .AND. (X(K) .LT. SPOSIT(J+
     *             1))) GO TO 125
140         CONTINUE
            IF((X(K) .GT. SPOSIT(NSING)) .AND. (X(K) .LT. A)) GO TO 126
121         SUM(1) = SUM(1) + AVNORM(K)
            GO TO 127
122         SUM(L+1) = SUM(L+1) + AVNORM (K)
            GO TO 127
123         SUM(N2+1) = SUM(N2+1) + AVNORM(K)
            GO TO 127
124         SUM(N3+1) = SUM(N3+1) + AVNORM(K)
            GO TO 127
125         SUM(J+2) = SUM(J+2) + AVNORM(K)
            GO TO 127
126         SUM(N1) = SUM(N1) + AVNORM(K)
127         CONTINUE
120 CONTINUE
293 CONTINUE
    DO 150 J=1,N1
            MCRECP(J) = IDINT((SUM(J)*(NPOINT-N1))/SUMT)
```

```
150 CONTINUE
    DELTA(1) = -(-A-SPOSIT(1))
    DO 260 J=1,N4
           DELTA(J+1) = -(SPOSIT(J) - SPOSIT(J+1))
260 CONTINUE
    DELTA(N3) = -SPOSIT(N2)
    DELTA(N3+1) = SPOSIT(N3)
    DO 270 J=N3,N5
           DELTA(J+2) = SPOSIT(J+1) - SPOSIT(J)
270 CONTINUE
    DELTA(N1) = A - SPOSIT(NSING)
    DO 280 J=1,N1
           MAXD(J) = IDINT(DELTA(J)*15.D0)
           NUMCP(J) = MORECP(J) + 1
           IF (NUMCP(J) .GT. MAXD(J)) NUMCP(J) = MAXD(J)
280 CONTINUE
    SUMNUM = 0.D0
    DO 290 J=1,N1
           SUMNUM = SUMNUM + NUMCP(J)
290 CONTINUE
    IF (SUMNUM .GT. NPOINT) GO TO 291
    GO TO 292
291 NPOINT = SUMNUM
    GO TO 293
292 NPOINT = SUMNUM
    Z1 = NUMCP(1)
    DO 180 K=1,Z1
           CPOSIT(K) = -A-(-A-SPOSIT(1))/(NUMCP(1)+1)*DFLOAT(K)
180 CONTINUE
    STOR7 = NUMCP(1)
```

```
      DO 190 K=1,N4
             STOR7 = STOR7 + NUMCP(K+1)
             M = STOR7 + 1 - NUMCP(K+1)
             CCUNT = 1
             DO 200 J=M,STOR7
                    CPOSIT(J) =SPOSIT(K) -(SPOSIT(K) -SPOSIT(K+1)) / (NUMCP(
     *              K+1)+1)*DFLOAT(CCUNT)
                    COUNT = COUNT + 1
200          CONTINUE
190   CONTINUE
      Z2 = NUMCP(N3)
      DO 210 K=1,Z2
             CPOSIT(STOR7+K) = SPOSIT(N2)/(NUMCP(N3+1)+1)*DFLOAT(K)
210   CONTINUE
      Z3 = NUMCP(N3+1)
      DO 220 K=1,Z3
             CPOSIT(STOR7+NUMCP(N3)+K) = SPOSIT(N3)/(NUMCP(N3+1)+1)*
     *       DFLOAT(K)
220   CONTINUE
      N6 = N2 + 2
      STOR8 = STOR7 + NUMCP(N3) + NUMCP(N3+1)
      DO 230 K=N6,NSING
             STOR8 = STOR8 + NUMCP(K+1)
             M1 = STOR8 + 1 - NUMCP(K+1)
             CCUNT1 = 1
             DO 240 J=M1,STOR8
                    CPOSIT(J) = SPOSIT(K-1)-(SPOSIT(K-1)-SPOSIT(K))/(
     *              NUMCP(K+1)+1)*DFLOAT(CCUNT1)
                    COUNT1 = COUNT1 + 1
240          CONTINUE
```

```
  230 CONTINUE
      M2 = STOR8 + 1
      COUNT2 = 1
      DO 250 K=M2,NPCINT
             CPOSIT(K) = SPOSIT(NSING)-((SPOSIT(NSING)-A)/(NUMCP(N1)+1)*
     *       DFLOAT(COUNT2))
             CCUNT2 = CCUNT2 + 1
  250 CONTINUE
*****************************************************************
* PASS NUMBER AND POSITION OF THE CCNTROL POINTS TO THE ADS OPTIMIZA-  *
* TION PROGRAM.                                                       *
*****************************************************************
      WRITE(3,301) NPCINT
      WRITE(3,300) (CPOSIT(I),I=1,NPCINT)
      IFL = 0
*****************************************************************
* CALCULATE AND FORM PLCTTING ARRAYS FOR ZERO STREAMLINE OF THE MODELED*
* BODY.                                                               *
*****************************************************************
      DO 90 KK=1,401
            XX = (-A - (2.D0*A/400.D0)) + (2.D0*A/400.D0)*DFLOAT(KK)
            W(KK) = DSQRT(1.D0 - (XX/A)**2)
  96        STORII = 0.D0
            DO 100 II=1,NSING
                  RII = DSQRT((((XX-SPOSIT(II))/A)**2)+(W(KK)**2*(B/A)
     *                 **2))
                  STORII = STORII+EE1(II,1)*((XX-SPOSIT(II))/A)/RII
  100       CONTINUE
            SI(KK) = (0.5D0*W(KK)**2)-STORII
            IF(DABS(SI(KK)).LT.0.01D0) GO TO 91
```

```
            JFL = IFL
            IF(SI(KK)) 92,91,93
   92       IFL = -1
            AK = .01D0
            GO TO 94
   93       IFL = 1
            AK = -.01D0
   94       IF(IFL+JFL) 95,91,95
   95       W(KK) = W(KK) + AK
            GO TO 96
   91       CONTINUE
            WPLCT(KK) = SNGL(W(KK))
            IFL = 0
   90 CONTINUE
*****************************************************************
* PLOTTING CALLS FOR NORMAL VELOCITY                            *
*****************************************************************
      CALL TEK618
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,8.5)
      CALL XNAME('X AXIS',6)
      CALL YNAME('NORMAL VELOCITY',15)
      CALL HEADIN('NORMAL VELOCITY VS X',-20,1.5,3)
      CALL HEADIN('A/E = 6.0',-9,1.5,3)
      CALL HEADIN('NO. SINGULARITIES = 20 NO. CONTROL PTS = 36',-43,1.5,
     *3)
      CALL GRAF(-7.0,1.0,7.0,-1.5,.25,1.5)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,VELOT,401,0)
```

```
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLOT,401,0)
      CALL CURVE(XPLCT,COPIOT,401,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
*****************************************************************
* PLOTTING CALLS FOR TANGENTIAL VELCCITY                        *
*****************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,8.5)
      CALL XNAME('X AXIS',6)
      CALL YNAME('TANGENTIAL VELOCITY',19)
      CALL HEADIN('TANGENTIAL VELOCITY VS X',-24,1.5,3)
      CALL HEADIN('A/B = 6.0',-9,1.5,3)
      CALL HEADIN('NO. SINGULARITIES = 20 NO. CONTROL PTS = 36',-43,1.5,
     *3)
      CALL GRAF(-7.0,1.0,7.0,-1.5,.25,1.5)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,VTPIOT,401,0)
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLOT,401,0)
      CALL CURVE(XPLCT,COPIOT,401,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
*****************************************************************
* PLOTTING CALLS FOR PRESSURE COEFFICIENT                       *
```

```
*****************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,8.5)
      CALL XNAME('X AXIS',6)
      CALL YNAME('PRESSURE COEFFICIENT',20)
      CALL HEADIN('PRESSURE COEFFICIENT VS X',-25,1.5,3)
      CALL HEADIN('A/E = 6.0',-9,1.5,3)
      CALL HEADIN('NO. SINGULARITIES = 20 NO. CONTROL PTS = 36',-43,1.5,
     *3)
      CALL GRAF(-7.0,1.0,7.0,-1.5,.25,1.5)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,CPPLOT,401,0)
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLOT,401,0)
      CALL CURVE(XPLCT,COPLOT,401,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
*****************************************************************
* PLOTTING CALLS FOR BODY SHAPE                                 *
*****************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,8.5)
      CALL XNAME('X AXIS',6)
      CALL YNAME('OMEGA',5)
      CALL HEADIN('BODY SHAPE',-10,1.5,3)
```

```
      CALL HEADIN('A/B = 6.0',-9,1.5,3)
      CALL HEADIN('NO. SINGULARITIES = 20 NO. CONTROL PTS = 36',-43,1.5,
     *3)
      CALL GRAF(-7.0,1.0,7.0,-1.5,.25,1.5)
      CALL CURVE(XPLCT,WPLOT,401,0)
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLOT,401,0)
      CALL CURVE(XPLCT,COPIOT,401,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
      CALL DONEPL
C ------------------------------------------------------------------
  300 FORMAT(F9.5)
  301 FORMAT(I3)
  400 FORMAT(2X,F9.5,2X,F9.5)
  500 FORMAT(2I4,2F9.5)
  501 FORMAT(F9.5)
  600 FORMAT(2X,'A/B = ',F9.5)
  601 FORMAT(2X,'SINGULARITY # ',I4,2X,'LOCATED AT X = ',F9.5)
  602 FORMAT(2X,'CONTROL POINT # ',I4,2X,'LOCATED AT X = ',F9.5)
  603 FORMAT(2X,'M(',I2,') = ',F9.5)
  604 FORMAT(1X)
  605 FORMAT(7X,'X',9X,'PSI',7X,'VNORM',6X,'VTANG',8X,'CP',9X,'U',9X,
     *'V')
  606 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5)
  607 FORMAT(2X,'S = ',F9.5)
  608 FORMAT(2X,'THE SUM OF THE STRENGTHS = ',F9.5)
      STOP
      END
```

```
**********************************************************************
*   THE FOLLOWING PROGRAM UTILIZES THE AUTOMATED DESIGN SYNTHESIS    *
* PROGRAM DEVELOPED BY G.N. VANDERPLAATS.  IT IS BEST TO REFER TO THE *
* DOCUMENTATION 'ADS - A FORTRAN PROGRAM FOR AUTOMATED DESIGN SYNTHE- *
* SIS' TO UNDERSTAND THE METHODOLOGY EMPLOYED TO WRITE THE PRESENT PRO-*
* GRAM.  PRIOR TO EXECUTING THE PRESENT PROGRAM, THE USER MUST LINK   *
* WITH THE ADS PROGRAM ON VANDERPLAATS' COMPUTER STORAGE DISK.        *
*   THE PROGRAM SOLVES FOR THE OPTIMUM POSITIONS OF THE SINGULARITIES *
* BASED ON THE INPUT DATA (REQUIRED INPUT DATA DENOTED IN VARIABLE LIST*
* THAT FOLLOWS.)                                                     *
********************** VARIABLES ***********************************
* A - HALF LENGTH OF MAJOR AXIS OF ELLIPSOID (INPUT)                 *
* B - HALF LENGTH OF MINOR AXIS OF ELLIPSOID (INPUT)                 *
* N - NUMBER OF SINGULARITIES (INPUT)                                *
* XOLD(I) - ARRAY OF THE POSITIONS OF THE SINGULARITIES ALONG THE X-  *
*           AXIS PRIOR TO OPTIMIZING THESE POSITIONS (INPUT)         *
* M(J) - ARRAY OF THE STRENGTHS OF THE SINGULARITIES RESULTING FROM THE*
*        GAUSS' LEAST SQUARES METHOD SOLUTION (INPUT)                *
* NPOINT - NUMBER OF CONTROL POINTS THAT RESULTED FROM CALCULATIONS IN *
*          THE PREVIOUS PROGRAM (INPUT)                              *
* CPOSIT(I) - ARRAY OF THE POSITIONS OF THE CONTROL POINTS ALONG THE  *
```

```
*             BODY SURFACE.  THESE POSITIONS RESULT FROM THE CALCULA-  *
*             TIONS IN THE PREVIOUS PROGRAM.  (INPUT)                 *
* NRA      NDV   ISTRAT                                               *
* NCOLA    NCON  IOPT      THESE VARIABLES ARE DESCRIBED IN THE ADS    *
* NRWK     VLB   IONED     DOCUMENTATION REFERENCED PREVIOUSLY.        *
* NRIWK    VUB   IPRINT                                               *
* IGRAD    IDG   INFO                                                 *
* BETA(I) - MOVE LIMITS IMPOSED ON THE POSITIONS OF EACH SINGULARITY IN*
*           A SINGLE ITERATION                                        *
* OBJ AND L - SUM OF THE SQUARES OF THE NORMAL VELOCITY AT A GIVEN NUM-*
*             BER OF POINTS ALONG THE BODY SURFACE.  THIS IS THE      *
*             QUANTITY THE ADS PROGRAM SEEKS TO MINIMIZE.             *
* G(J)  - ARRAY OF THE CONSTRAINTS SET FOR THE MOVEMENT OF EACH DESIGN *
*         VARIABLE (I.E. THE POSITIONS OF THE SINGULARITIES)          *
* X(P)  - ARRAY OF THE NEW OPTIMUM POSITIONS OF THE SINGULARITIES     *
* XK(J) - ARRAY OF POINTS ALONG THE BODY SURFACE (DIFFERENT FROM THE  *
*         CONTROL POINTS) AT WHICH THE NORMAL VELOCITIES ARE CALCULATED*
* WK(I) - RADIAL DISTANCE FROM X-AXIS TO A GIVEN XK(I) POINT          *
* RK - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN XK(I) POINT *
* U - VELOCITY COMPONENT IN THE X-DIRECTION AT A GIVEN XK(I) POINT    *
* V - VELOCITY COMPONENT IN THE  -DIRECTION AT A GIVEN XK(I) POINT    *
* ALPHA - ANGLE BETWEEN U AND THE TANGENTIAL VELOCITY COMPONENT AT A  *
*         GIVEN XK(I) POINT                                           *
* VNORM - NORMAL VELOCITY COMPONENT AT A GIVEN XK(I) POINT            *
**********************************************************************
      INTEGER I,II,J,K,KK,N,NN,NNN,JJ,KKK,N1,N2,NPOINT,K1
      REAL A,B,L,STOR1,STOR2,STOR3,WK,RK,U,V,ALPHA,VNORM,M,X,BETA,XOLD
      REAL DELTA,XK,CPOSIT
      DIMENSION BETA(50),VLB(50),VUB(50),G(30),IDG(30),IC(30),DF(50),
     *AA(50,40),WK(25000),IWK(10000),M(50),X(50),XOLD(50),CPOSIT(50),
```

```
     *XK(50)
*************************************************************************
* INPUT AXIS LENGTHS,NUMBER AND POSITION OF SINGULARITIES AND CONTROL   *
* POINTS, AND STRENGTHS OF SINGULARITIES.                               *
*************************************************************************
      READ(5,500)  A
      READ(5,500)  B
      READ(5,501)  N
      READ(5,500)  (XOLD(I),I=1,N)
      READ(5,500)  (M(J),J=1,N)
      READ(5,501)  NPOINT
      READ(5,500)  (CPOSIT(I),I=1,NPOINT)
*************************************************************************
* ECHO INPUT DATA TO OUTPUT FILE AND PASS TO LEAST SQUARES METHOD       *
* PROGRAM.                                                              *
*************************************************************************
      WRITE(6,600)  A,B
      WRITE(6,605)  N
      WRITE(3,300)  N,NPOINT,A,B
      DO 30 II=1,N
            WRITE(6,601)  II,XOLD(II)
   30 CONTINUE
      DO 40 K=1,N
            WRITE(6,602)  K,M(K)
   40 CONTINUE
*************************************************************************
* INITIALIZE REQUIRED INPUT FOR CALLING ADS.                            *
*************************************************************************
      NN = N/2
      NNN = N-1
```

```
      NRA = 50
      NCCLA = 40
      NRWK = 25000
      NRIWK = 10000
      IGRAD = C
      NDV = N
      NCON = N + 2
      DC 10 I=1,N
            BETA(I) = 0.0
            VLB(I) = -3.14159265/2.C
            VUB(I) = 3.14159265/2.0
   10 CONTINUE
      DO 11 KKK=1,NCCN
            IDG(KKK) = 0
   11 CONTINUE
      ISTRAT = 8
      IOPT = 3
      ICNED = 9
      IPRINT = 2200
      INFO = 0
*****************************************************************
* CALL THE ADS SUBROUTINE AND THE ANALYSIS SUBROUTINE           *
*****************************************************************
   20 CALL ADS(INFO,ISTRAT,IOPT,IONED,NDV,NCON,IPRINT,IGRAD,BETA,VLB,
     *VUB,OBJ,G,IDG,NGT,IC,DF,AA,NRA,NCOLA,WK,NRWK,IWK,NRIWK)
      CALL ANALIZ(INFO,BETA,A,B,N,NNN,M,XOLD,NPOINT,CPOSIT,X,L,XK,K1)
      OBJ = L
*****************************************************************
* ESTABLISH THE EQUATIONS FOR THE CCNSTRAINTS                   *
*****************************************************************
```

```
      G(1) = -A-X(1)
      DO 60 J=2,NN
            G(J) = X(J-1)-X(J)
   60 CONTINUE
      G(NN+1) = X(NN)
      G(NN+2) = -X(NN+1)
      N1 = NN+3
      N2 = N+1
      DC 70 JJ=N1,N2
            G(JJ) = X(JJ-2)-X(JJ-1)
   70 CCNTINUE
      G(N+2) = X(N)-A
      IF (INFO.GT.0) GO TO 20
*****************************************************************
* WRITE RESULTS IN OUTPUT FILE AND PASS TO LEAST SQUARES METHOD PROGRAM*
*****************************************************************
      WRITE(6,*) L
      DO 50 KK=1,N
            WRITE(6,603) KK,X(KK)
            WRITE(3,301) X(KK)
   50 CONTINUE
      WRITE(3,301) (CPOSIT(I),I=1,NPCINT)
      WRITE(6,301) (XK(I),I=1,K1)
  300 FCRMAT(2I4,2F9.5)
  301 FORMAT(F9.5)
  500 FCRMAT(F9.5)
  501 FCRMAT(I3)
  600 FCRMAT(2X,'A = ',F9.5,2X,'B = ',F9.5)
  601 FORMAT(2X,'XOLD(',I2,') = ',F9.5)
  602 FCRMAT(2X,'M(',I2,') = ',F9.5)
```

```
  603 FORMAT(2X,'X(',I2,') = ',F9.5)
C 604 FORMAT(2X,'L = ',F9.5)
  605 FORMAT(2X,'NUMBER OF SINGULARITIES = ',I3)
      STOP
      END
*************************************************************************
* THIS SUBROUTINE PERFORMS CALCULATIONS FOR THE OPTIMUM POSITIONS OF    *
* THE SINGULARITIES AND THE OBJECTIVE FUNCTION MINIMIZED BY ADS.        *
*************************************************************************
      SUBROUTINE ANALIZ(INFO,BETA,A,E,N,NNN,M,XOLD,NPOINT,CPOSIT,X,L,XK
     *,K1)
      DIMENSION BETA(50),XK(50),M(50),X(50),XOLD(50),CPOSIT(50)
      REAL A,B,L,STOR1,STOR2,STOR3,WK,RK,U,V,ALPHA,VNORM,M,XK,X,XOLD
      REAL BETA,DELTA1,CPOSIT
      INTEGER I,J,K1,P,NNN,N,JJ,JJJ,LLL,N5,N6,NPOINT,COUNT
           X(1)=((XOLD(1)+XOLD(2))/2.0-A)/2.0+((XOLD(1)+XOLD(2))/2.0+A)
     *          /2.0*SIN(BETA(1))
           DO 30 P=2,NNN
                X(P)=(((XOLD(P)+XOLD(P+1))/2.0)+((XOLD(P-1)+XOLD(P))/
     *               2.0))/2.0+(((XOLD(P)+XOLD(P+1))/2.0)-((XOLD(P-1)+
     *               XOLD(P))/2.0))/2.0*SIN(BETA(P))
   30      CONTINUE
           X(N)=(A+((XOLD(N-1)+XOLD(N))/2.0))/2.0+(A-((XOLD(N-1)+XOLD(N
     *          ))/2.0))/2.0*SIN(BETA(N))
           K1 = NPOINT + 2
           COUNT = 0
           DO 31 J=1,NPOINT
                IF (CPOSIT(J) .LT. 0.0) COUNT = COUNT + 1
   31      CONTINUE
           XK(1) = (-A + CPOSIT(1))/2.0
```

```
      DO 40 JJ=2,CCUNT
            XK(JJ) = (CPOSIT(JJ-1) + CPOSIT(JJ))/2.0
40    CONTINUE
      XK(CCUNT+1) = CPOSIT(COUNT)/2.0
      XK(CCUNT+2) = CPOSIT(COUNT+1)/2.0
      N5 = COUNT + 3
      N6 = K1 - 1
      DO 50 JJ=N5,N6
            XK(JJ) = (CPOSIT(JJ-2) + CPOSIT(JJ-1))/2.0
50    CONTINUE
      XK(K1) = (A + CPOSIT(NPOINT))/2.0
      STOR3 = 0.0
      DO 10 I=1,K1
            STOR1 = 0.0
            STOR2 = 0.0
            WK =  SQRT(1.0-(XK(I)/A)**2)
            DO 20 J=1,N
                  RK = SQRT(((XK(I)-X(J))/A)**2)+((WK**2)*(B/A)
     *            **2))
                  STOR1 = STOR1+M(J)*(((XK(I)-X(J))/A)/RK**3)*(B/A
     *            )**2
                  STOR2 = STOR2+M(J)*((WK/RK**3)*(B/A)**3)
20          CONTINUE
            U = 1.0 + STOR1
            V = STOR2
            ALPHA = ATAN(-(XK(I)/A)/WK*(E/A))
            VNORM = V*COS(ALPHA) - U*SIN(ALPHA)
            STOR3 = STOR3 + VNORM**2
10    CONTINUE
      L = STOR3
```

```
      RETURN
      END
```

```
**********************************************************************
*   THE FOLLOWING PROGRAM IS USED TO MODEL POTENTIAL FLOW ABOUT AXISYM-*
* METRIC BODIES WITHOUT FORE AND AFT SYMMETRY USING DISCRETE SINGULAR- *
* ITIES.  IT PERFORMS THE FOLLOWING:
*     1) USING THE GAUSS LEAST SQUARES METHOD, SOLVES FOR THE STRENGTHS*
*        OF THE SINGULARITIES.                                         *
*     2) CALCULATES THE FLOW CHARACTERISTICS ABOUT THE BODY(I.E. V ,V ,*
*        C ).                                                          *
*     3) APPORTIONS CONTROL POINTS ALONG THE BODY SURFACE AS NEEDED.   *
*     4) PLOTS THE FLOW CHARACTERISTICS ABOUT THE BODY.                *
*   INPUT DATA REQUIRED IS DENOTED IN THE LIST OF VARIABLES THAT       *
* FOLLOWS.                                                            *
********************** VARIABLES ***********************************
* NSING - NUMBER OF SINGULARITIES (INPUT)                             *
* NPOINT - NUMBER OF CONTROL POINTS (INPUT)                           *
* SPOSIT(I) - ARRAY OF THE POSITIONS OF THE SINGULARITIES ALONG THE X- *
*             AXIS (INPUT)                                            *
* CPOSIT(J) - ARRAY OF THE POSITIONS OF THE CONTROL POINTS ALONG THE   *
*             BODY SURFACE (INPUT)                                    *
* OMEGA(J) - RADIAL DISTANCE FROM MAJOR AXIS TO A CONTROL POINT (INPUT)*
* X(K) - ARRAY OF EQUALLY SPACED POINTS ALONG THE BODY SURFACE (INPUT) *
* OMEGA1(K) - RADIAL DISTANCE FROM MAJOR AXIS TO X(K) POINT (INPUT)    *
* ALPHA(K) - ANGLE BETWEEN U AND THE TANGENTIAL VELOCITY COMPONENT AT A*
*            X(K) POINT (INPUT)                                       *
* BB(J,1) - ARRAY OF TERMS IN STOKES' STREAM FUNCTION EQUATION CONTRI- *
*           BUTED BY THE FREE STREAM VELOCITY                         *
* R - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN CONTROL POINT*
* AA(J,I) - ARRAY OF TERMS IN THE STOKES STREAM FUNCTION EQUATION CON- *
*           TRIBUTED BY THE SINGULARITIES                             *
* BB1(I,1) - ARRAY OF THE STRENGTHS OF THE SINGULARITIES              *
```

```
* R1 - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN X(K) POINT  *
* PSI - VALUE OF THE STOKES' STREAM FUNCTION AT A GIVEN X(K) POINT     *
* U - VELOCITY COMPONENT IN THE X-DIRECTION AT A GIVEN X(K) POINT      *
* V - VELOCITY COMPONENT IN THE  -DIRECTION AT A GIVEN X(K) POINT      *
* VNORM(K) - ARRAY OF THE NORMAL VELOCITY COMPONENTS AT THE X(K) POINTS*
* AVNORM(K) - ABSOLUTE VALUE OF VNORM(K)                              *
* SUMT - TOTAL SUM OF AVNORM(K) ALONG THE BODY SURFACE                *
* VTANG - ARRAY OF THE TANGENTIAL VELOCITY COMPONENTS AT THE X(K)     *
*         POINTS                                                      *
* CP - PRESSURE COEFFICIENT AT A GIVEN X(K) POINT                     *
* S - AVERAGE ROOT MEAN SQUARE VALUE OF THE NORMAL VELOCITIES         *
* SUM(K) - ARRAY OF SUM OF AVNORM(K) FOR A GIVEN INTERVAL ALONG THE   *
*          BODY SURFACE                                               *
* MORECP(K) - ARRAY OF NUMBER OF ADDITIONAL CONTROL POINTS APPORTIONED *
*             TO A PARTICULAR INTERVAL ALONG THE BODY SURFACE          *
* DELTA(J) - ARRAY OF INTERVAL LENGTHS ALONG THE BODY SURFACE          *
* MAXD(J) - ARRAY OF MAXIMUM NUMBER OF CONTROL POINTS THAT CAN BE AP-  *
*           PORTIONED TO A GIVEN INTERVAL (BASED ON A MAXIMUM DENSITY  *
*           CRITERIA OF 15 CONTROL POINTS PER UNIT LENGTH)             *
* NUMCP(J) - NUMBER OF CONTROL POINTS ACTUALLY APPORTIONED TO A GIVEN  *
*            INTERVAL AFTER A COMPARISON OF MORECP(J) AND MAXD(J) FOR  *
*            THAT INTERVAL                                            *
* SUMNUM - TOTAL SUM OF ALL NUMCP(J)                                  *
* RII - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN X(KK) POINT*
* SI(KK) - VALUE OF STOKES' STREAM FUNCTION AT A GIVEN X(KK) POINT     *
**********************************************************************
      REAL*8 OMEGA(50),B,OMEGA1(700),STOR1,PSI,U,V,ALPHA(700),STORII
      REAL*8 STOR2,STOR3,R1,STOR4,S,RII,AK,SIGMAM,SUMT,AVNORM(900)
      REAL*8 SPOSIT(50),CPOSIT(50),BB(50,1),AA(50,50),BB1(50,1),SUM(50)
      REAL*8 WKAREA(900),SI(900),AA1(50,50),X(900),VNORM(900),DIFF
```

```
      REAL*8 DELTA(50),CP,VTANG
      REAL*4 XPLOT(900),YPLOT(900),OPLOT(900),OOPLOT(900),VTPLOT(900)
      REAL*4 CPPLOT(900),WPLOT(900)
      INTEGER I,J,K,NSING,NPOINT,KK,II,IFL,JFL,N1,N5,MORECP(50)
      INTEGER STOR7,STOR8,M,M2,NUMCP(50),N2
      INTEGER Z1,COUNT,COUNT2,MAXD(50),SUMNUM
*************************************************************
* INPUT AXIS LENGTHS, NUMBER AND POSITION OF CONTROL POINTS AND SINGU- *
* LARITIES. INPUT FROM BODY TO BE MODELED ARRAY OF EQUALLY SPACED      *
* POINTS ALONG THE BODY SURFACE AND ARRAYS OF THE RESPECTIVE OMEGA AND *
* ALPHA VALUES AT EACH POINT.                                          *
*************************************************************
      READ(5,500) NSING,NPOINT
      READ(5,501) (SPOSIT(I),I=1,NSING)
      READ(5,501) (CPOSIT(J),J=1,NPOINT)
      READ(5,501) (OMEGA(J),J=1,NPOINT)
      READ(5,502) (X(K),OMEGA1(K),ALPHA(K),K=1,602)
*************************************************************
* PASS DATA TO ADS OPTIMIZATION PROGRAM                                *
*************************************************************
      WRITE(3,302) (X(K),OMEGA1(K),ALPHA(K),K=1,602)
      WRITE(3,301) NSING
      WRITE(3,300) (SPOSIT(I),I=1,NSING)
*************************************************************
* ECHO NUMBER AND POSITION OF SINGULARITIES AND CONTROL POINTS TO OUT- *
* PUT FILE                                                             *
*************************************************************
      DO 10 I=1,NSING
            WRITE(6,601) I,SPOSIT(I)
 10   CONTINUE
```

```
      DO 20 J=1,NPOINT
            WRITE(6,602) J,CPOSIT(J)
   20 CONTINUE
*****************************************************************
* SOLVE FOR STRENGTHS OF SINGULARITES FROM EQUATIONS OF THE STREAM FUNC*
* TION AT EACH CONTROL POINT.                                    *
*****************************************************************
      DO 30 J=1,NPOINT
            BB(J,1) = 0.5D0*OMEGA(J)**2
            DO 40 I=1,NSING
                  R = DSQRT(((CPOSIT(J)-SPOSIT(I))**2) + (OMEGA(J)**2))
                  AA(J,I) =(CPOSIT(J)-SPOSIT(I))/R
   40       CONTINUE
   30 CONTINUE
*****************************************************************
* INITIALIZE INPUT DATA NECESSARY FOR CALLING LIBRARY SUBROUTINES *
* VMULFM AND LEQT2F.                                             *
*****************************************************************
      N = 1
      IA = 50
      IDGT = 10
      CALL VMULFM(AA,AA,NPOINT,NSING,NSING,IA,IA,AA1,IA,IER)
      CALL VMULFM(AA,EE,NPOINT,NSING,N,IA,IA,BB1,IA,IER)
      CALL LEQT2F(AA1,N,NSING,IA,BB1,IDGT,WKAREA,IER)
      SIGMAM = C.D0
*****************************************************************
* WRITE VALUES OF THE STRENGTHS OF THE SINGULARITIES IN OUTPUT FILE AND*
* PASS TO ADS OPTIMIZATION PROGRAM.                              *
*****************************************************************
      DO 50 I=1,NSING
```

```
            WRITE(6,603) I,BB1(I,1)
            WRITE(3,300) BB1(I,1)
            SIGMAM = SIGMAM + BB1(I,1)
 50   CONTINUE
      WRITE(6,608) SIGMAM
      WRITE(6,604)
      WRITE(6,605)
*****************************************************************
* CALCULATE THE STREAM FUNCTION AT A MULTITUDE OF EQUALLY SPACED POINTS*
* ALONG THE BODY. ESTABLISH PLOTTING ARRAYS FOR AXISYMMETRIC BODY BEING*
* MODELED.                                                      *
*****************************************************************
      STOR4 = C.D0
      SUMT = 0.D0
      DO 60 K = 1,602
            XPLOT(K) = SNGL(X(K))
            OPLOT(K) = SNGL(OMEGA1(K))
            OOPLOT(K) = -OPLOT(K)
            STOR1 = 0.D0
            DO 70 I =1,NSING
                  R1 = DSQRT(((X(K)-SPOSIT(I))**2)+(OMEGA1(K)**2))
                  STOR1 = STOR1 + BB1(I,1)*(X(K)-SPOSIT(I))/R1
 70         CONTINUE
            PSI = (0.5D0*OMEGA1(K)**2) - STOR1
*****************************************************************
* CALCULATE NORMAL VELOCITY, TANGENTIAL VELOCITY, AND THE PRESSURE CO- *
* EFFICIENT AT EACH (X,OMEGA) POINT ON THE BODY SURFACE.  ESTABLISH    *
* PLOTTING ARRAYS FOR SAME.                                     *
*****************************************************************
            STOR2 = 0.D0
```

```
            STOR3 = 0.D0
            DO 80 I=1,NSING
                  R1 = DSQRT(((X(K)-SPOSIT(I))**2)+(OMEGA1(K)**2))
                  STOR2 = STOR2 + BE1(I,1)*((X(K)-SPOSIT(I))/R1**3)
                  STOR3 = STOR3 + BE1(I,1)*(OMEGA1(K)/R1**3)
 80         CONTINUE
            U = 1.D0 + STOR2
            V = STOR3
            VNORM(K) = V*DCOS(ALPHA(K)) - U*DSIN(ALPHA(K))
            STOR4 = STOR4 + VNORM(K)**2
            AVNORM(K) = DABS(VNORM(K))
            SUMT = SUMT + AVNORM(K)
            VPLOT(K) = SNGL(VNORM(K))
            VTANG = U*DCOS(ALPHA(K)) + V*DSIN(ALPHA(K))
            WRITE(1,300) VTANG
            VTPLOT(K) = SNGL(VTANG)
            CP = 1.D0 - (U**2 + V**2)
            CPPLOT(K) = SNGL(CP)
            WRITE(6,606) X(K),PSI,VNORM(K),VTANG,CP,U,V
            WRITE(4,400) X(K),VTANG
 60   CONTINUE
*************************************************************************
* CALCULATE THE AVERAGE ROOT MEAN SQUARE VALUE OF THE NORMAL VELOCITY   *
*************************************************************************
      S = DSQRT(STOR4)/602.D0
      WRITE(6,606) S
*************************************************************************
* DIVIDE THE BODY INTO INTERVALS BOUNDED BY SUCCESSIVE SINGULARITIES.   *
* APPORTION TO EACH INTERVAL THE NUMBER OF CONTROL POINTS REQUIRED      *
* BASED ON THE RATIO OF THE SUM OF THE ABSOLUTE VALUES OF THE NORMAL    *
```

```
* VELOCITIES OF A PARTICULAR INTERVAL TO THE TOTAL SUM OF THE ABSOLUTE *
* VALUES OF THE NORMAL VELCCITIES FCR THE WHOIE BODY.                  *
************************************************************************
      N1 = NSING + 1
      DC 110 K=1,N1
             SUM(K) = 0.DO
  110 CCNTINUE
      N5 = NSING - 1
      DC 120 K=1,602
             IF((X(K) .GT. 0.0) .AND. (X(K) .IT. SPOSIT(1))) GO TO 121
             DC 130 I=1,N5
                    IF((X(K) .GT. SECSIT(L)) .AND. (X(K) .LT. SPOSIT(L
     *              +1))) GO TO 122
  130        CONTINUE
             IF((X(K) .GT. SPOSIT(NSING)) .AND. (X(K) .LT. 10.74)) GO
     *       TO 126
  121        SUM(1) = SUM(1) + AVNOBM(K)
             GC TO 127
  122        SUM(L+1) = SUM(L+1) + AVNORM (K)
             GC TO 127
  126        SUM(N1) = SUM(N1) + AVNCBM(K)
  127        CONTINUE
  120 CONTINUE
  293 CCNTINUE
      DO 150 J=1,N1
             MCRECP(J) = IDINT( (SUM(J)*(NPOINT-N1))/SUMT)
  150 CONTINUE
      DELTA(1) = SPOSIT(1)
      DO 260 J=1,N5
             DELTA(J+1) = (SPOSIT(J+1) - SPOSIT(J))
```

```
260 CCNTINUE
    DELTA(N1) = 10.74 - SPOSIT(NSING)
    DC 280 J=1,N1
           MAXD(J) = IDINT(DELTA(J)*15.D0)
           NUMCP(J) = MCRECP(J) + 1
           IF (NUMCP(J) .GT. MAXD(J)) NUMCP(J) = MAXD(J)
280 CONTINUE
    SUMNUM = 0.D0
    DO 290 J=1,N1
           SUMNUM = SUMNUM + NUMCP(J)
290 CONTINUE
    IF (SUMNUM .GT. NPOINT) GO TO 291
    GO TO 292
291 NPCINT = SUMNUM
    GO TO 293
292 NPCINT = SUMNUM
    Z1 = NUMCP(1)
    DC 180 K=1,Z1
           CPOSIT(K) = (SPOSIT(1)/(NUMCP(1)+1))*DFLOAT(K)
180 CCNTINUE
    N2 = NSING -1
    STCR7 = NUMCP(1)
    DO 190 K=1,N2
           STOR7 = STCR7 + NUMCP(K+1)
           M = STOR7 + 1 - NUMCP(K+1)
           CCUNT = 1
           DC 200 J=M,STCR7
                  CPOSIT(J) =SPOSIT(K)+(SPOSIT(K+1)-SPOSIT(K))/(NUMCP(
   *              K+1)+1)*DFLOAT(CCUNT)
                  COUNT = COUNT + 1
```

```
  200        CONTINUE
  190 CONTINUE
      M2 = STOR7 + 1
      COUNT2 = 1
      DO 250 K=M2,NPOINT
             CPOSIT(K) = SPOSIT(NSING)+((10.74-SPOSIT(NSING))/(NUMCP(N1)
     *       +1)*DFLOAT(COUNT2))
             COUNT2 = COUNT2 + 1
  250 CONTINUE
*****************************************************************
* FIND X(K) POINT (AND OMEGA(K) CORRESPONDING) FROM INPUT DATA CLOSEST *
* TO CPOSIT(K) CALCULATED.  USE THESE X(K) POINTS AS THE NEW CONTROL   *
* POINTS.                                                              *
*****************************************************************
      DO 1 J=1,NPOINT
           DO 2 K=1,602
                DIFF = DABS(CPOSIT(J)-X(K))
                IF (DIFF .LE. .00895) GO TO 3
  2        CONTINUE
           GO TO 4
  3        CPOSIT(J) = X(K)
           OMEGA(J) = OMEGA1(K)
  4        CONTINUE
  1   CONTINUE
*****************************************************************
* PASS NUMBER OF CONTROL POINTS, THEIR POSITIONS AND OMEGA VALUES TO   *
* ADS OPTIMIZATION PROGRAM.                                            *
*****************************************************************
      WRITE(3,301) NPOINT
      WRITE(3,300) (CPOSIT(I),I=1,NPOINT)
```

```
      WRITE(3,300)  (CMEGA(J),J=1,NPCINT)
      IFL = 0
*************************************************************
* CALCULATE AND FORM ELCTTING ARRAYS FOR ZERO STREAMLINE OF MODELED BODY
*************************************************************
      DO 90 KK=1,602
 96         STORII = 0.D0
            DO 100 II=1,NSING
                  RII = DSQRT(((X(KK)-SPOSIT(II))**2)+(OMEGA1(KK)**2))
                  STORII = STORII+EE1(II,1)*(X(KK)-SPOSIT(II))/RII
100         CONTINUE
            SI(KK) = (0.5D0*OMEGA1(KK)**2)-STORII
            IF(DABS(SI(KK)).LT.0.01D0) GO TO 91
            JFL = IFL
            IF(SI(KK)) 92,91,93
 92         IFL = -1
            AK = .01D0
            GO TO 94
 93         IFL = 1
            AK = -.01D0
 94         IF(IFL+JFL) 95,91,95
 95         OMEGA1(KK) = OMEGA1(KK) + AK
            GO TO 96
 91         CONTINUE
            WPLOT(KK) = SNGL(OMEGA1(KK))
            IFL = 0
 90   CONTINUE
*************************************************************
* PLOTTING CALLS FOR NORMAL VELOCITY                        *
*************************************************************
```

```
      CALL TEK618
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,4.0)
      CALL XNAME('X AXIS',6)
      CALL YNAME('NORMAL VELOCITY',15)
      CALL GRAF(0.0,1.0,11.0,-2.0,.50,2.0)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,VPLOT,602,0)
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLOT,602,0)
      CALL CURVE(XPICT,COPIOT,602,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
C*****************************************************************
C* PLOTTING CALLS FOR TANGENTIAL VELOCITY                        *
C*****************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,4.0)
      CALL XNAME('X AXIS',6)
      CALL YNAME('TANGENTIAL VELOCITY',19)
      CALL GRAF(0.0,1.0,11.0,-2.0,.50,2.0)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,VTPIOT,602,0)
      CALL CHNDSH
      CALL CURVE(XPICT,CPLOT,602,0)
      CALL CURVE(XPICT,COPIOT,602,0)
```

```
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
*********************************************************************
* PLOTTING CALLS FOR PRESSURE COEFFICIENT                           *
*********************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,4.0)
      CALL XNAME('X AXIS',6)
      CALL YNAME('PRESSURE COEFFICIENT',20)
      CALL GRAF(0.0,1.0,11.0,-2.0,.50,2.0)
      CALL GRID(0,1)
      CALL CURVE(XPLOT,CPPLOT,602,0)
      CALL CHNDSH
      CALL CURVE(XPLCT,CPLCT,602,0)
      CALL CURVE(XPLCT,COPLOT,602,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
*********************************************************************
* PLOTTING CALLS FOR BODY SHAPE                                     *
*********************************************************************
      CALL RESET('ALL')
      CALL PAGE(13.,10.5)
      CALL NOBRDR
      CALL AREA2D(11.,4.0)
      CALL XNAME('X AXIS',6)
      CALL YNAME('OMEGA',5)
```

```
      CALL GRAF(0.0,1.0,11.0,-2.0,.50,2.0)
      CALL CURVE(XPLOT,WPLOT,602,0)
      CALL CHNDSH
      CALL CURVE(XPLOT,CPLOT,602,0)
      CALL CURVE(XPLOT,COPLOT,602,0)
      CALL FRAME
      CALL THKFRM(.02)
      CALL ENDPL(0)
      CALL DONEPL
C ------------------------------------------------------------------
  300 FORMAT(F9.5)
  301 FORMAT(I3)
  302 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5)
  400 FORMAT(2X,F9.5,2X,F9.5)
  500 FORMAT(2I4,2F9.5)
  501 FORMAT(F9.5)
  502 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5)
  600 FORMAT(2X,'A/E = ',F9.5)
  601 FORMAT(2X,'SINGULARITY # ',I4,2X,'LOCATED AT X = ',F9.5)
  602 FORMAT(2X,'CONTROL POINT # ',I4,2X,'LOCATED AT X = ',F9.5)
  603 FORMAT(2X,'M(',I2,') = ',F9.5)
  604 FORMAT(1X)
  605 FORMAT(7X,'X',9X,'PSI',7X,'VNORM',6X,'VTANG',8X,'CP',9X,'U',9X,
     *'V')
  606 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5,2X,F9.5)
  607 FORMAT(2X,'S = ',F9.5)
  608 FORMAT(2X,'THE SUM OF THE STRENGTHS = ',F9.5)
      STOP
      END
```

```
C**********************************************************************
C*   THE FOLLOWING PROGRAM UTILIZES THE AUTOMATED DESIGN SYNTHESIS PRO- *
C* GRAM DEVELOPED BY G.N. VANDERPLAATS.  IT IS BEST TO REFER TO THE DOC-*
C* UMENTATION 'ADS - A FORTRAN PROGRAM FOR AUTOMATED DESIGN SYNTHESIS'  *
C* TO UNDERSTAND THE METHODOLOGY EMPLOYED TO WRITE THE PRESENT PROGRAM. *
C* PRIOR TO EXECUTING THE PRESENT PROGRAM THE USER MUST BE LINKED WITH  *
C* THE ADS PROGRAM ON VANDERPLAATS' STORAGE DISK.                       *
C*   THE PROGRAM SOLVES FOR THE OPTIMUM POSITIONS OF THE SINGULARITIES  *
C* BASED ON THE INPUT DATA (REQUIRED INPUT DATA DENOTED IN LIST OF VAR- *
C* IABLES THAT FOLLOWS).                                               *
C*************************** VARIABLES **********************************
C* XORIG(K) - ARRAY OF EQUALLY SPACED POINTS ALONG THE BODY SURFACE (IN-*
C*            PUT)                                                     *
C* OMEGA1(K) - RADIAL DISTANCE FROM MAJOR AXIS TO XORIG(K) POINT ON BODY*
C*             SURFACE (INPUT)                                         *
C* ALPHA(K) - ANGLE BETWEEN U AND THE TANGENTIAL VELOCITY COMPONENT AT A*
C*            GIVEN XORIG(K) POINT (INPUT)                             *
C* N - NUMBER OF SINGULARITIES (INPUT)                                 *
C* XOLD(I) - ARRAY OF THE POSITIONS OF THE SINGULARITIES ALONG THE X-   *
C*           AXIS PRIOR TO OPTIMIZING THEIR POSITIONS (INPUT)          *
C* M(J) - ARRAY OF THE STRENGTHS OF THE SINGULARITIES RESULTING FROM THE*
C*        GAUSS' LEAST SQUARES METHOD SOLUTION (INPUT)                 *
C* NPOINT - NUMBER OF CONTROL POINTS THAT RESULTED FORM THE CALCULATIONS*
C*          IN THE PREVIOUS PROGRAM (INPUT)                            *
C* CPOSIT(I) - ARRAY OF THE POSITIONS OF THE CONTROL POINTS ALONG THE   *
C*             BODY SURFACE.  THESE POSITIONS RESULT FROM CALCULATIONS  *
C*             IN THE PREVIOUS PROGRAM.                                *
C* OMEGA(I) - RADIAL DISTANCE FROM X-AXIS TO A GIVEN CONTROL POINT      *
C* NEA     NDV    ISTRAT                                               *
C* NCOLA   NCON   IOPT      THESE VARIABLES ARE BEST UNDERSTOOD BY      *
```

```
* NRWK    VLB    IONED    REFERENCING THE DOCUMENTATION DESCRIBED     *
* NRIWK   VUB    IPRINT   PREVIOUSLY                                  *
* IGRAD   IDG    INFC                                                *
* BETA(I) - MOVE LIMITS IMPOSED ON THE POSITIONS OF THE SINGULARITIES *
*           IN ONE ITERATION                                         *
* OBJ AND L - SUM OF THE SQUARES OF THE NORMAL VELOCITY AT A GIVEN NUM-*
*             BER OF POINTS ALONG THE BODY SURFACE.  THIS IS THE QUAN- *
*             TITY THE ADS PROGRAM IS MINIMIZING.                      *
* X(P) - ARRAY OF THE NEW OPTIMUM POSITIONS OF THE SINGULARITIES       *
* XK(J) - ARRAY OF POINTS ALONG THE BODY SURFACE (DIFFERENT THAN THE   *
*         CONTROL POINTS INPUTTED) AT WHICH THE NORMAL VELOCITY IS     *
*         CALCULATED                                                   *
* OMEGAK(J) - RADIAL DISTANCE FROM X-AXIS TO XK(J) POINT               *
* RK - RADIAL DISTANCE FROM A GIVEN SINGULARITY TO A GIVEN XK(J) POINT *
* U - VELOCITY COMPONENT IN THE X-DIRECTION AT A GIVEN XK(J) POINT     *
* V - VELOCITY COMPONENT IN THE  -DIRECTION AT A GIVEN XK(J) POINT     *
* ALPHA(J) - ANGLE BETWEEN U AND THE TANGENTIAL VELOCITY COMPONENT AT A*
*          _  GIVEN XK(J) POINT                                        *
* VNORM - NORMAL VELOCITY COMPONENT AT A GIVEN XK(J) POINT             *
**********************************************************************
      INTEGER I,J,K,KK,N,JJ,KKK,NPOINT,K1,II,N1
      REAL L,STOR1,STOR2,STOR3,RK,U,V,VNORM,DIFF
      REAL BETA(50),VLB(50),VUB(50),G(30),IDG(30),IC(30),DF(50),
     *AA(50,40),WK(25000),IWK(10000),M(50),X(50),XOLD(50),CPOSIT(50),
     *XK(50),XORIG(700),OMEGA1(700),ALPHA(700),OMEGAK(50),OMEGA(50)
**********************************************************************
* INPUT FROM BODY TO BE MODELED AN ARRAY OF EQUALLY SPACED POINTS ALONG*
* THE BODY SURFACE AND ARRAYS OF THE RESPECTIVE OMEGA AND ALPHA VALUES *
* AT EACH POINT.  INPUT NUMBER, POSITION, AND STRENGTH OF SINGULARITIES*
* AND NUMBER, POSITION, AND OMEGA VALUES OF CONTROL POINTS.            *
```

```
*****************************************************************
      READ(5,502)  (XCRIG(K),OMEGA1(K),ALPHA(K),K=1,602)
      READ(5,501)  N
      READ(5,500)  (XOLD(I),I=1,N)
      READ(5,500)  (M(J),J=1,N)
      READ(5,501)  NPCINT
      READ(5,500)  (CPOSIT(I),I=1,NPCINT)
      READ(5,500)  (OMEGA(I),I=1,NPOINT)
*****************************************************************
* ECHO INPUT DATA TO OUTPUT FILE AND PASS TO LEAST SQUARES METHOD PRO- *
* GRAM.                                                               *
*****************************************************************
      WRITE(6,605)  N
      WRITE(3,300)  N,NPCINT
      DO 30 II=1,N
            WRITE(6,601) II,XOLD(II)
   30 CONTINUE
      DO 40 K=1,N
            WRITE(6,602) K,M(K)
   40 CONTINUE
*****************************************************************
* INITIALIZE REQUIRED INPUT FOR CALLING ADS                          *
*****************************************************************
      NRA = 50
      NCOLA = 40
      NRWK = 25000
      NRIWK = 10000
      IGRAD = 0
      NDV = N
      NCON = N + 1
```

```
      N1 = N - 1
      DO 10 I=1,N
            BETA(I) = 0.0
            VLB(I) = -3.14159265/2.0
            VUB(I) = 3.14159265/2.0
   10 CONTINUE
      DO 11 KKK=1,NCON
            IDG(KKK) = 0
   11 CONTINUE
      ISTRAT = 8
      IOPT = 3
      ICNED = 9
      IPRINT = 2200
      INFO = 0
*******************************************************************
* CALL THE ADS SUBROUTINE AND THE ANALYSIS SUBROUTINE             *
*******************************************************************
   20 CALL ADS(INFO,ISTRAT,IOPT,IONED,NDV,NCON,IPRINT,IGRAD,BETA,VLB,
     *VUB,OBJ,G,IDG,NGT,IC,DF,AA,NRA,NCOLA,WK,NRWK,IWK,NRIWK)
      CALL ANALIZ(INFO,XCRIG,OMEGA1,ALPHA,BETA,N,M,XOLD,NPOINT,CPOSIT,
     *N1,X,L,XK,K1)
      OBJ = L
*******************************************************************
* ESTABLISH EQUATIONS FOR THE CONSTRAINTS                         *
*******************************************************************
      G(1) = -X(1)
      DO 60 J=2,N
            G(J) = X(J-1)-X(J)
   60 CONTINUE
      G(N+1) = X(N) - 10.74
```

```
      IF (INFO.GT.0) GO TO 20
*****************************************************************
* WRITE RESULTS IN OUTPUT FILE AND PASS TO LEAST SQUARES METHOD PROGRAM*
*****************************************************************
      WRITE(6,*) L
      DO 50 KK=1,N
            WRITE(6,603) KK,X(KK)
            WRITE(3,301) X(KK)
   50 CONTINUE
      WRITE(3,301)  (CPOSIT(I),I=1,NECINT)
      WRITE(6,301)  (XK(I),I=1,K1)
      WRITE(3,301)  (CMEGA(J),J=1,NPOINT)
      WRITE(3,302)  (XORIG(K),OMEGA1(K),ALPHA(K),K=1,602)
  300 FORMAT(2I4,2F9.5)
  301 FORMAT(F9.5)
  302 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5)
  500 FORMAT(F9.5)
  501 FORMAT(I3)
  502 FORMAT(2X,F9.5,2X,F9.5,2X,F9.5)
  600 FORMAT(2X,'A = ',F9.5,2X,'B = ',F9.5)
  601 FORMAT(2X,'XOLD(',I2,') = ',F9.5)
  602 FORMAT(2X,'M(',I2,') = ',F9.5)
  603 FORMAT(2X,'X(',I2,') = ',F9.5)
C 604 FORMAT(2X,'L = ',F9.5)
  605 FORMAT(2X,'NUMBER OF SINGULARITIES = ',I3)
      STOP
      END
*****************************************************************
* THIS SUBROUTINE PERFORMS CALCULATIONS FOR NEW POSITIONS OF THE SINGU-*
* LARITIES AND THE OBJECTIVE FUNCTION MINIMIZED BY ADS.              *
```

```
**********************************************************************
      SUBROUTINE ANALIZ(INFO,XORIG,OMEGA1,ALPHA,BETA,N,M,XOLD,NPOINT,
     *CPOSIT,N1,X,L,XK,K1)
      DIMENSION BETA(50),XK(50),M(50),X(50),XOLD(50),CPOSIT(50),XORIG(
     *700),OMEGA1(700),ALPHA(700),OMEGAK(50),ALPHAK(50)
      REAL L,STOR1,STOR2,STOR3,RK,U,V,ALPHA,VNORM,M,XK,X,XOLD,OMEGAK
      REAL BETA,CPOSIT,DIFF,ALPHAK
      INTEGER I,J,K1,P,N1,N,JJ,JJJ,ILL,NPOINT
           X(1)=((XOLD(1)+XOLD(2))/2.0)/2.0+((XOLD(1)+XOLD(2))/2.0)
     *          /2.0*SIN(BETA(1))
           DO 30 P=2,N1
                X(P)=(((XOLD(P)+XOLD(P+1))/2.0)+((XOLD(P-1)+XOLD(P))/
     *               2.0))/2.0+(((XCLD(P)+XOLD(P+1))/2.0)-((XOLD(P-1)+
     *               XCLD(P))/2.0))/2.0*SIN(BETA(P))
30         CONTINUE
           X(N)=(10.74+((XOLD(N-1)+XOLD(N))/2.0))/2.0+(10.74-((XOLD(N-
     *     1)+XOLD(N))/2.0))/2.0*SIN(BETA(N))
           K1 = NPOINT + 1
           XK(1) = (CPOSIT(1))/2.0
           DO 40 JJ=2,NPOINT
                XK(JJ) = (CPOSIT(JJ-1) + CPOSIT(JJ))/2.0
40         CONTINUE
           XK(K1) = (10.74 + CPOSIT(NPOINT))/2.0
           STOR3 = 0.0
           DO 1 J=1,K1
                DO 2 K=1,602
                     DIFF =ABS(XK(J)-XORIG(K))
                     IF(DIFF .LE. .00895) GO TO 3
2               CONTINUE
                GO TO 4
```

```
3           XK(J) = XORIG(K)
            OMEGAK(J) = OMEGA1(K)
            ALPHAK(J) = ALPHA(K)
4           CONTINUE
1       CONTINUE
        DO 10 I=1,K1
            STOR1 = 0.0
            STOR2 = 0.0
            DO 20 J=1,N
                RK = SQRT(((XK(I)-X(J))**2)+(OMEGAK(I)**2))
                STOR1 = STOR1+M(J)*((XK(I)-X(J))/RK**3)
                STOR2 = STOR2+M(J)*(OMEGAK(I)/RK**3)
20          CONTINUE
            U = 1.0 + STOR1
            V = STOR2
            VNORM = V*COS(ALPHAK(I)) - U*SIN(ALPHAK(I))
            STOR3 = STOR3 + VNORM**2
10      CONTINUE
        L = STOR3
        RETURN
        END
```

INITIAL DISTRIBUTION LIST

		No. Copies
1.	Defense Technical Information Center Cameron Station Alexandria, Virginia 22314	2
2.	Department Chairman, Code 69 Department of Mechanical Engineering Naval Postgraduate School Monterey, California 93943	1
3.	Library, Code 0142 Naval Postgraduate School Monterey, California 93943	2
4.	Professor T. Sarpkaya, (Code 69Sl) Department of Mechanical Engineering Monterey, California 93943	5
5.	Lieutenant Linda C. Janikowsky, USN 43 Pamela Court Spotswood, New Jersey 08884	5

206776

Thesis
J2853
c.1

Janikowsky
Representation of potential flow about axisymmetric bodies with discrete singularities.